AVOCADO: Adaptive Optimal Collision Avoidance Driven by Opinion

Diego Martinez-Baselga, *Graduate Student Member, IEEE*, Eduardo Sebastián, *Graduate Student Member, IEEE*, Eduardo Montijano, *Member, IEEE*, Luis Riazuelo, *Member, IEEE*, Carlos Sagüés, *Senior Member, IEEE*, and Luis Montano, *Senior Member, IEEE*

***Abstract*—We present AdaptiVe Optimal Collision Avoidance Driven by Opinion (AVOCADO), a novel navigation approach to address holonomic robot collision avoidance when the robot does not know how cooperative the other agents in the environment are. AVOCADO departs from a velocity obstacle's (VO) formulation akin to the optimal reciprocal collision avoidance method. However, instead of assuming reciprocity, it poses an adaptive control problem to adapt to the cooperation level of other robots and agents in real time. This is achieved through a novel nonlinear opinion dynamics design that relies solely on sensor observations. As a by-product, we leverage tools from the opinion dynamics formulation to naturally avoid the deadlocks in geometrically symmetric scenarios that typically suffer VO-based planners. Extensive numerical simulations show that AVOCADO surpasses existing motion planners in mixed cooperative/noncooperative navigation environments in terms of success rate, time to goal and computational time. In addition, we conduct multiple real experiments that verify that AVOCADO is able to avoid collisions in environments crowded with other robots and humans.**

***Index Terms*—Collision avoidance, motion and path planning, multirobot systems, opinion dynamics.**

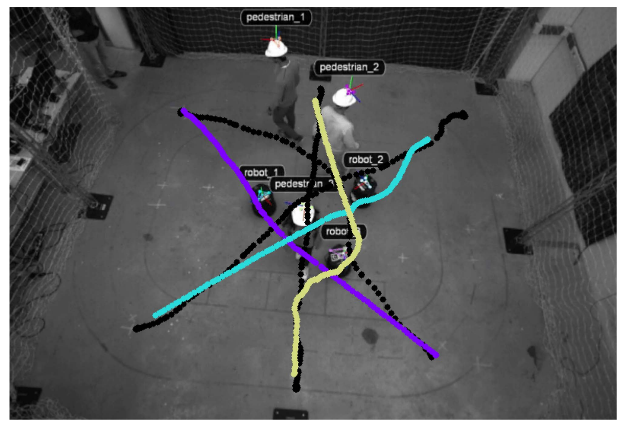


Fig. 1. Illustrative example of one of the experiments with real robots and humans. Robots using AVOCADO adapt online to avoid collisions with the other entities in the arena despite not knowing the degree of cooperation of the other robots and humans. Section VI discusses all the experiments in detail.

I. Introduction

ROBOT navigation in dynamic environments is the task of moving a robot from one location to another while avoiding collisions with obstacles that are in motion around the environment [1]. Collision avoidance is particularly challenging when the dynamic obstacles are agents (humans, other robots, etc., as in Fig. 1), and the problem of accounting for the effect of the robot motion on the other agents is still unsolved [2]. Agents have their own intent and they exhibit different degrees of cooperation, i.e., they adjust their behavior to avoid collision with the robot to a certain extent that is unknown to the robot. Thus, it is hard for robots to predict their future motion and effectively plan to avoid potential collisions [3], [4]. The ability of avoiding collisions in crowded dynamic environments is a low-level feature that is of key importance in robotic applications, such as social robotics [5], [6], robots in the wild [7], [8], [9], or autonomous driving [10], [11].

Existing solutions (see Section II) make one or more of the following assumptions.

1) The degree of cooperation between the robot and an agent is known a priori (e.g., the robot knows that the agent is going to collaborate in the collision avoidance and how), restricting the solution to predetermined cooperative settings [12], [13], [14], [15].
2) The robot can communicate with other robots and/or agents [16], [17], [18], [19], which might not be possible during deployment due to a lack of cooperation, communication delays, packet losses or disturbances from external factors.
3) The computational capabilities of the robot are sufficient to do inference on learning-based policies or long-term horizon planning [20], [21], [22], [23], which is hard to ensure in low-cost onboard platforms if we take into

Received 11 December 2024; accepted 20 February 2025. Date of publication 17 March 2025; date of current version 7 April 2025. This work was supported in part by MICIU/AEI/10.13039/501100011033, ERDF/EU, in part by the European Union NextGenerationEU/PRTR and ESF Investing in your future under Grant PID2022-139615OB-I00, Grant PID2021-125514NB-I00, Grant PID2021-124137OB-I00, Grant TED2021-130224B-I00, Grant PRE2020-094415, Grant FPU19-05700, in part by the Government of Aragón under Grant DGA T45-23R, in part by the ONR Global under Grant N62909-24-1-2081. This article was recommended for publication by Associate Editor W. Yao and Editor P. Robuffo Giordano upon evaluation of the reviewers' comments. *(Diego Martinez-Baselga and Eduardo Sebastián contributed equally to this work.) (Corresponding author: Diego Martinez-Baselga.)*

The authors are with the Instituto de Investigación en Ingeniería de Aragón (I3A), Universidad de Zaragoza, 500018 Zaragoza, Spain (e-mail: diegomartinez@unizar.es; esebastian@unizar.es; emonti@unizar.es; riazuelo@unizar.es; csagues@unizar.es; montano@unizar.es).

This article has supplementary downloadable material available at https://doi.org/10.1109/TRO.2025.3552350, provided by the authors.

Digital Object Identifier 10.1109/TRO.2025.3552350

© 2025 The Authors. This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

account that collision avoidance is just a low-level feature among a myriad of desired high-level and computationally demanding capabilities (e.g., multirobot active perception or semantic mapping [24], [25]).

To cope with all these issues, **the main contribution** of this work is **A**daptiVe **O**ptimal **C**ollision **A**voidance **D**riven by **O**pinion (AVOCADO), a novel method for collision avoidance in cooperative and noncooperative dynamic environments (see Section IV). AVOCADO is built upon a geometrical formulation based on the classical velocity obstacle (VO, [26], [27]) framework (see Section III). However, different from other VO-based approaches that assume either no cooperation or reciprocity [12], [28], [29], we consider that the degree of cooperation between the robot and the agent is unknown and do not assume any predetermined behavior for the agent or communication infrastructure. To address uncertainty on the degree of cooperation, we propose a novel nonlinear opinion dynamics adaptive scheme that enables the robot to adapt to the agent's behavior in real-time from just its onboard sensor observations. Furthermore, we exploit the nonlinear opinion dynamics adaptive law to develop a method that avoids the geometrical deadlocks coming from symmetries that suffer some existing VO-based methods. We validate our proposal with simulations (see Section V) and real experiments (see Section VI), showing that AVOCADO surpasses exiting collision avoidance approaches in environments that mix cooperative and noncooperative agents in terms of avoided collisions while featuring an inexpensive computational cost. The scenarios include examples that combine real robots and humans in tight spaces, concluding (see Section VII) that AVOCADO is a promising alternative for low-cost robot navigation in crowded environments where agents exhibit unknown degrees of cooperation.

The code of the work is implemented in C++/Python and can be accessed in the repository of the work,[1] as well as the videos of the experiments.

II. Related Work

Research on collision avoidance was primarily dominated by geometrical approaches. Departing from robot observations, methods originally only considered static obstacles, as the dynamic window approach [30]. Others such as potential fields [31], inevitable collision states [32], or social forces [33], compute control actions that reactively lead the robot toward collision-free regions. They are simple to compute but lack formal performance guarantees, leading to overconservative policies. By contrast, VO-based methods [34] preserve the computational simplicity but guarantee optimality in terms of deviation with respect to the desired predefined velocity either for holonomic or nonholonomic robots [29], [35], [36]. Reciprocal VO (RVO) extends VO by assuming reciprocity in the collision avoidance [28], i.e, it assumes that agents encountered in the environment cooperate equally to avoid the collision. Especially relevant is optimal reciprocal collision avoidance ORCA [12], which builds a half-space geometrical constraint per agent that is incorporated in a linear program. Nevertheless, the reciprocal avoidance assumption may not be applied in mixed cooperative/noncooperative environments, which are the ones considered in this article.

Some solutions [37] remove the reciprocity assumption in ORCA and formulate an additional optimization problem to reduce the time the agents take to reach the goal. Instead of using optimization, ORCA can also be implemented as a low-level module integrated in a high-level learned policy that accounts for social and traffic cues [38]. Both approaches require one-hop communication, which is not feasible in a non-cooperative environment due to the lack of cooperation among agents. Instead, AVOCADO is fully decentralized and only relies on the perception of the robot.

Many recent methods use deep learning for robot navigation. Either from a supervised [20], [39], [40], [41], [42] or reinforcement learning perspective [21], [43], [44], [45], [46], [47], [48], deep learning develop end-to-end policies that map the perception observations into control commands [14] or subgoals that are fed into a low-level horizon planning controller [49]. For instance, long short-term memory networks can be combined with barrier certificates [50] to improve collision avoidance success rate and scalability. There are also works that combine ORCA scenario knowledge with reinforcement learning [38], [51]. Nonetheless, deep learning approaches have a high memory and computational cost, often lack collision avoidance guarantees and fail in out-of-distribution settings.

The lack of guarantees of purely learning-based methods is addressed by reachability methods [16], control barrier functions [52] and model predictive control (MPC) [53]. Reachability-based approaches depart from the Hamilton–Jacobi equation [54] to build a safe set in the state space that fulfills all the desired constrains. Forward computation of the Hamilton–Jacobi equation is highly demanding and overconservative [55], so learning-based alternatives [56], [57], [58] propose approximations that trade computation and performance guarantees. On the other hand, control barrier functions formulate an optimization problem that finds the closest control action to the desired one that satisfies a set of desired forward invariance constraints, stemming from a control stability definition of the desired safe set. It is not clear how to extend control barrier functions to dynamic obstacles with unknown behaviors [59] or communication-denied settings [60], [61]. Finally, MPC shares with reachability methods the use of a time-horizon propagation and with control barrier functions the optimization-based formulation [62]. MPC is flexible because it enables to satisfy additional goals or constraints, such as motion uncertainties [63] or local maps [22]. However, it has a significant computational burden that is only alleviated by reducing the time-horizon and the number of considered nearby agents, resulting in undesired local minima or oscillations. Different from these alternatives, AVOCADO is inexpensive to compute, is not over-conservative as it minimizes the deviation with respect to the desired velocity, and does not suffer from symmetry deadlocks and oscillations.

[1] [Online]. Available: https://github.com/dmartinezbaselga/AVOCADO

One key characteristic of the aforementioned methods is that, in noncooperative environments with unknown intents, a prediction model of the behavior of the other agents is required to effectively avoid collisions [64], [65]. The first option is to use learned predictors that directly obtain future trajectory states of the agents given current and past samples [66], [67]. The second option is to rely on a learned model [68], such that it can be integrated in an MPC program [23] to simultaneously optimize over the trajectories of the robot and the other agents. In both cases, the success of the collision avoidance module is subject to the accuracy of the complex prediction modules; instead, AVOCADO adapts to unknown degrees of cooperation through a novel nonlinear opinion dynamics adaptive law that is inexpensive to compute, does not require learning and is only based on current perception observations, without relying on prediction of future trajectories.

Opinion dynamics [69], [70], [71] originate from studies on how to model social interactions. The ineffectiveness of linear models to represent behaviors, such as saturation of information leads to nonlinear opinion dynamics [72], [73]. They have been applied in a wide variety of fields, such as explaining political polarization [74] or perception and reaction to epidemics [75]; lately, they have been applied in robotic problems, mainly combining them with game theory for cooperative and noncooperative multiagent decision making [76], [77], [78] or multirobot task allocation [79]. Regarding collision avoidance, there is an important property of nonlinear opinion dynamics. They ensure the existence of a bifurcation in the state–space, i.e., the existence of two simultaneous stable equilibrium points. This feature has been exploited [80] to design a collision avoidance method that avoids deadlocks, where the opinion represents the preference of the robot to move left or right. A recent work [81] extends opinion dynamics from the planar to the circle space to represent, e.g., the heading angle of a robot. Nevertheless, these approaches do not guarantee collision avoidance nor consider other motion patterns than only changing the heading angle. AVOCADO exploits nonlinear opinion dynamics to guarantee that the robot decides if the agent is cooperative or noncooperative and in what degree, leaving the collision avoidance guarantees to a VO-based program.

III. Problem Formulation

Consider a robot navigating in an environment populated with $\mathsf{N} > 0$ agents. The robot follows single integrator holonomic dynamics given by

$$\dot{\mathbf{p}}_r = \mathbf{v}_r^{\mathsf{pre}} \tag{1}$$

where $\mathbf{p}_r \in \mathbb{R}^2$ is the position of the robot and $\mathbf{v}_r^{\mathsf{pre}} \in \mathbb{R}^2$ is a prescribed velocity command generated by a higher level motion planner. On the other hand, each agent is associated with an index $i \in \{1, \dots, \mathsf{N}\}$, and has a position $\mathbf{p}_i \in \mathbb{R}^2$ and a velocity $\mathbf{v}_i \in \mathbb{R}^2$. The robot can only sense the agents that are within a disc centered in the robot position and of perception radius $r_p > 0$

$$\mathsf{D}_p(\mathbf{p}_r, r_p) = \{\mathbf{p} \mid ||\mathbf{p} - \mathbf{p}_r|| < r_p\} \tag{2}$$

where $|| \bullet ||$ is the L2-norm of a vector and $\mathbf{p} \in \mathbb{R}^2$. Therefore, the set of neighboring agents of the robot is defined as $\mathcal{N} = \{i \mid \mathbf{p}_i \in \mathsf{D}_p(\mathbf{p}_r, r_p)\}$. The purpose of the article is to develop an algorithm that avoids collisions between the robot and the agents. To that end, we define the set of locations that lead to collisions as a disc centered in the position of the robot and security radius $r_s > 0$

$$\mathsf{D}_c(\mathbf{p}_r, r_s) = \{\mathbf{p} \mid ||\mathbf{p} - \mathbf{p}_r|| < r_s\} \tag{3}$$

where r_s is given by the geometry of the robot and safety requirements. Agents are also characterized by a collision radius $r_i > 0$. The degree of cooperation of agent i with respect to the robot is modeled through a continuous variable $\alpha_i \in [0, 1]$, where $\alpha_i = 1$ means that the agent is fully cooperative with the robot (i.e., agent i will do all the efforts at its hands to avoid collision), whereas $\alpha_i = 0$ means that the agent is completely noncooperative (i.e., agent i will ignore the robot and continue its movement without any collision avoidance effort). We assume that agents are not competitive, i.e., agents do not try to force collision with the robot as in, e.g., pursuit-evasion [82] or herding [83] problems. As in realistic mixed cooperative/noncooperative environments, the degree of cooperation of the agents is unknown.

To model collisions, we exploit the concept of VO [26], [27]. Given a certain time instant $t \geq 0$, the VO of the robot and agent i is the set of velocities that can lead to a collision

$$\mathsf{VO}_i = \{\mathbf{v}_r \mid \exists t \in [0, \tau] \text{ s.t. } t\mathbf{v}_r \in \mathsf{D}_c(\mathbf{p}_i - \mathbf{p}_r, r_s + r_i)\}. \tag{4}$$

In VO_i, $\tau > 0$ is the time horizon to check collisions between the robot and agent i happens for any velocity $\mathbf{v}_r \in \mathbb{R}^2$. In Section IV, we make use of VO_i to develop an optimization program that uses the adapted degree of cooperation to compute a velocity $\mathbf{v}_r^*$ as close as possible to $\mathbf{v}_r^{\mathsf{pre}}$ that guarantees collision avoidance.

To adapt to the unknown degree of cooperation, we rely on the concept of nonlinear opinion dynamics [72]. Opinion dynamics emerge as a means of modeling interactive behaviors where nodes in a network can choose among a discrete set of options regarding one or more topics. Let $o_j \in [-1, 1]$ be the opinion that node j has about a certain topic. Typically, $o_j = 0$ represents neutrality whereas $o_j = \{-1, 1\}$ represent extreme opinions over a topic with two possible opinions. By interacting with other nodes, node j can evolve its opinion. For a topic with two opinions, one possible nonlinear opinion dynamics design is

$$\dot{o}_j = -d_j o_j + g(o_j) f\left(a_j o_j + \sum_{k=1, k\neq j}^{\mathsf{M}} c_{jk} o_k\right) + b_j. \tag{5}$$

In the above-mentioned equation, $d_j > 0$ tunes how fast the current opinion vanishes with time; $b_j \in \mathbb{R}$ is a bias that models inherent priorities that the node j has on the topic; $g(o_j)$ is an attention term that evolves with time and weights the importance of the influence of other nodes; $a_j > 0, c_{jk} > 0$ are consensus-like weights that model the exchange of opinions among the $\mathsf{M} > 0$ nodes of the network; and $f(\bullet)$ is a nonlinear function, such as tanh or sigmoid, that bounds the influence of the interactions. More importantly, the nonlinear function in the opinion dynamics enforces a bifurcation phenomenon. The bifurcation phenomenon is characterized by the appearance of a pitchfork bifurcation in the evolution of the state variable o_j that leads to two nonzero stable equilibrium, each of them associated to one of the opinions. Whether the state evolves to one or another depends on the interactions between nodes. For more details on this, we refer the reader to [72]. In relation with this work, the inherent bifurcation property of nonlinear opinion dynamics and the conceptual relationships between the opinion and the parameters of (5) motivate their use to estimate ("build an opinion") on the unknown degree of cooperation of the other agents.

In this work, we will exploit nonlinear opinion dynamics to design a novel adaptive law that adjusts the unknown degree of cooperation of the agents in real-time, using only relative position and velocity measurements from onboard sensors and without the need of any communication infrastructure, prior knowledge on their degree of cooperation or other unfeasible assumptions. As a practical note, all the expressions that involve continuous-time dynamic systems are implemented using an Euler forward discretization scheme, with sample time $T > 0$. In this work, we select $T = 0.05$ s to match the real-time operation of the onboard sensing and computation capabilities of the robot.

IV. AVOCADO: Adaptive Optimal Collision Avoidance Driven by Opinion

A. Optimal Collision Avoidance for Unknown Degrees of Cooperation

We propose AVOCADO to address the robot collision avoidance problem in multiagent dynamic environments. The first step toward deriving AVOCADO is to consider the geometry defined by VO_i for all $i \in \{1, \ldots, \mathsf{N}\}$. Let assume that the relative velocity $\mathbf{v}_r^{\mathsf{pre}} - \mathbf{v}_i \in \mathsf{VO}_i$ for some agent i. This means that, if robot and agent i follow $\mathbf{v}_r^{\mathsf{pre}}$ and $\mathbf{v}_i$, respectively, then collision will happen at any time in $[0, \tau]$. The velocity $\mathbf{v}_r^{\mathsf{pre}}$ is provided by some higher level planner, while $\mathbf{v}_i$ is known from onboard sensor measurements. The latter can be measured or estimated in real-time from onboard cameras or LiDAR [84], [85]. Since cooperation of agent i cannot be assumed, then the robot needs to choose a new velocity $\mathbf{v}_r^*$ to avoid the collision.

Let $\partial\mathsf{VO}_i$ denote the boundary of the VO set VO_i. Then, the minimum change in velocity that makes $\mathbf{v}_r^{\mathsf{pre}} - \mathbf{v}_i \notin \mathsf{VO}_i$ is given by the vector $\mathbf{u}_i$

$$\mathbf{u}_i = \underset{\mathbf{v}_r \in \partial\mathsf{VO}_i}{\arg\min} ||\mathbf{v}_r - (\mathbf{v}_r^{\mathsf{pre}} - \mathbf{v}_i)|| - (\mathbf{v}_r^{\mathsf{pre}} - \mathbf{v}_i). \tag{6}$$

Fig. 2 depicts the geometrical reasoning behind (6). Given the Euclidean space of 2-D velocities, vector $\mathbf{u}_i$ is a perpendicular vector to the boundary of the VO set and with minimum magnitude. To ensure collision avoidance, the robot and agent i together must, at least, exert vector $\mathbf{u}_i$. Let α_i be the degree of cooperation of agent i with respect to the robot such that agent i exerts the α_i part of $\mathbf{u}_i$. Then, to ensure collision avoidance, the robot has to exert, at least, the $(1 - \alpha_i)$ part of $\mathbf{u}_i$, which defines

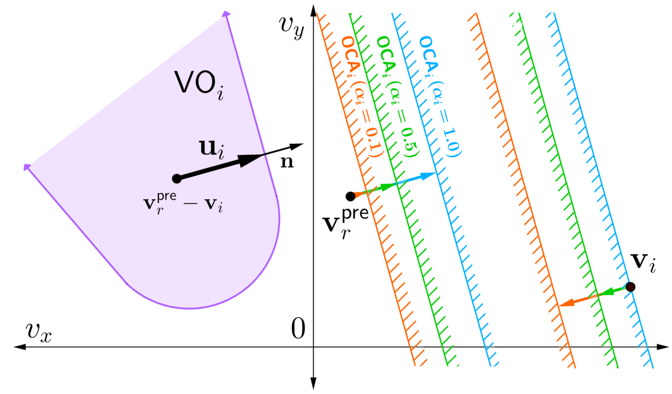


Fig. 2. Geometry of VO_i (purple) and admissible velocities' regions associated to different degrees of cooperation (orange, green, blue). AVOCADO selects the closest velocity to the desired one that is inside all the admissible velocity sets.

a constraint set of admissible velocities

$$\mathsf{OCA}_i = \{\mathbf{v}_r \mid (\mathbf{v}_r - (\mathbf{v}_r^{\mathsf{pre}} + (1 - \alpha_i)\mathbf{u}_i)) \cdot \mathbf{n} \geq 0\} \tag{7}$$

where $\mathbf{n}$ is the normal to $\partial\mathsf{VO}_i$ at the point given by the head of vector $\mathbf{u}_i$. The set OCA_i defines a half-space constraint of admissible velocities for the robot to avoid collision. As depicted in Fig. 2, the greater the value of α_i, the more cooperative is agent i and therefore the less severe the restriction on the admissible velocities for the robot. Note that $\alpha_i = 0.5$ corresponds to the ORCA algorithm [12], which corresponds to perfect reciprocity between the robot and agent i.

Given OCA_i for all $i \in \{1, \ldots, \mathsf{N}\}$, AVOCADO selects the closest velocity to $\mathbf{v}_r^{\mathsf{pre}}$ that respects all the constraints sets at the same time. Formally, this is given by the following linear program:

$$\mathbf{v}_r^* = \underset{\mathbf{v}_r}{\arg\min} \quad ||\mathbf{v}_r^{\mathsf{pre}} - \mathbf{v}_r|| \tag{8a}$$

$$\text{s.t.} \quad \mathbf{v}_r \in \mathsf{OCA}_i \quad \forall i \in \mathcal{N}. \tag{8b}$$

Problem (8) is guaranteed to have a solution unless the intersection of the OCA_i sets is empty. In this case, the robot computes the velocity that minimizes the distance with the half-planes $\mathsf{OCA}_i \quad \forall i \in \mathcal{N}$, i.e., it chooses the "least unsafe" velocity in the absence of a velocity that guarantees collision avoidance in that instant (see Section 5.3 of [12] for further details). We remark that (8) is a natural extension of ORCA to settings where the degree of cooperation of the agents is not necessarily reciprocal. Fig. 3 shows a representation of the constrained set (8b).

The challenge is how to estimate α_i for all perceived agents $i \in \mathcal{N}$ from local sensing. Next, we describe how AVOCADO exploits opinion dynamics to adapt online to the degree of cooperation of each agent.

B. Nonlinear Opinion Dynamics Adaptive Law

To derive the adaptive law based on nonlinear opinion dynamics, we first define, for convenience, the following change

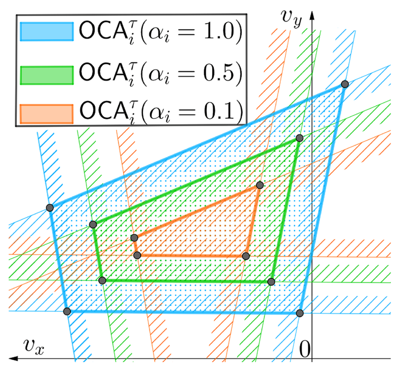


Fig. 3. Representation of the intersection set specified by the OCA_i half-planes of four agents for different values of α_i. Note that α_i may be different for each of the agents, but it is considered the same in this figure for a simple visualization. If the intersection set of all half-planes is not empty, the problem is guaranteed to have a solution.

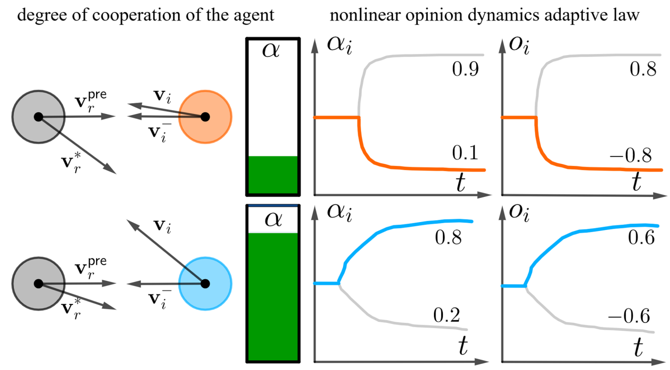


Fig. 4. Evolution of the nonlinear opinion dynamics adaptive law for a noncooperative (top, orange) or a cooperative (bottom, blue) agent.

of coordinates:

$$o_i = 2\alpha_i - 1 \Leftrightarrow \alpha_i = \frac{o_i + 1}{2}. \tag{9}$$

The purpose of this change is to shift the degree of cooperation to better suit the definition of nonlinear opinion dynamics provided in (5). From now on, we refer to o_i as the shifted degree of cooperation of agent i with respect to the robot.

The (shifted) degree of cooperation associated to each agent is a quantity that only involves a pairwise interaction between the robot and the agent. This motivates the following design for the nonlinear opinion dynamics adaptive law:

$$\dot{o}_i = -d_i o_i + d_i A_i \mathtt{tanh}\left(a_i o_i + c_i e_i\right) + b_i. \tag{10}$$

Compared to (5), (10) simplifies the consensus term to a weighted sum of the current shifted degree of cooperation o_i and a quantity e_i that will be defined later and which represents an estimate of the opinion agent i has about o_i. Ideally, the consensus term would include not only e_i, but also all the other agents. However, this would require some sort of communication infrastructure or k-hop neighboring dependence that is not feasible in a real setting that only depends on onboard sensing. Therefore, we assume that the opinion of the robot on the degree of cooperation of agent i is a pairwise interaction isolated from other agents. In this sense, the design of (10) manifests a fundamental difference with respect to existing nonlinear opinion dynamics formulations, tailored toward a feasible implementation in a real robotic platform. The quantities a_i, b_i, c_i, d_i are gains with similar meaning to those in (5) and which parameterized the adaptive law. For instance, $b_i = -0.5$ represents some inductive bias on the degree of cooperation of agent i toward not cooperating with the robot. As nonlinear function, we choose $\mathtt{tanh}$ to respect the domain of o_i. Finally, A_i represents the attention, and is a variable that dynamically evolves with time, in contrast to the algebraic relationship $g(o_j)$ formulated in (5). As depicted in Fig. 4, the idea is to design and tune (10) such that, when the agent i is close to the robot and does not collaborate to avoid the collision, the (shifted) degree of cooperation associated to agent i evolves such that the robot takes the responsibility. On the other hand, if agent i modifies its velocity to avoid collision, then the (shifted) degree of cooperation evolves to enable the robot a more selfish behavior. The attention term A_i appears scaled in (10) by gain d_i. The reason behind this scaling will be explained later, in Section IV-C.

There are two variables in (10) that are not readily available at the robot. The first one is the attention, which should capture the importance of adapting the shifted degree of cooperation or if, otherwise, the shifted degree of cooperation must not change. The second quantity is e_i, which a priori depends on the information at agent i. We will later prove how the robot can estimate e_i from just onboard sensor observations. Now, we focus on the attention A_i.

The goal is to design an attention mechanism for the nonlinear opinion dynamics adaptive law that evolves such that the shifted degree of cooperation changes when the robot and agent i are close to each other, whereas it remains constant if the robot and the agent are far. The attention A_i is modeled as a dynamic quantity that evolves with time

$$\dot{A}_i = -\delta_i A_i + (1 - \delta_i)\mathtt{tanh}(\kappa_i \tau_i^{-1}) \tag{11}$$

where $\kappa_i > 0$ and $\delta_i \in [0, 1)$ are gains and $\tau_i > 0$ is the expected collision time. The dynamics in (11) evolve such that $A_i = 0$ when the robot and agent i are far from each other, and A_i tends to 1 when the robot and agent i are close to collision. To compute the time to collision, we exploit the geometry of the problem.

A potential collision between the robot and agent i happens at the intersection between 1) a circumference of radius $R = r_s + r_i$ and center $\mathbf{p}_i$ and 2) the line given by $\mathbf{v}_r^{\mathrm{pre}} - \mathbf{v}_r$ passing through point $\mathbf{p}_r$. Let $\mathbf{p}_r = (p_r^x, p_r^y)$ and $\mathbf{p}_i = (p_i^x, p_i^y)$. The equation of the circumference with center $\mathbf{p}_i$ and radius R is

$$(p^x - p_i^x)^2 + (p^y - p_i^y)^2 = R^2 \tag{12}$$

where the circumference is parameterized by $\mathbf{p} = (p^x, p^y)$. Let $\tilde{\mathbf{v}}_i = \mathbf{v}_r^{\mathrm{pre}} - \mathbf{v}_i = (\tilde{v}_i^x, \tilde{v}_i^y)$. Then, the intersection between 1) the circumference defined in (12) and 2) a line of slope given by vector $\tilde{\mathbf{v}}_i$ and point $\mathbf{p}_r$ happens at

$$(p_r^x + \tau_i \tilde{v}_i^x - p_i^x)^2 + (p_r^y + \tau_i \tilde{v}_i^y - p_i^y)^2 = R^2. \tag{13}$$

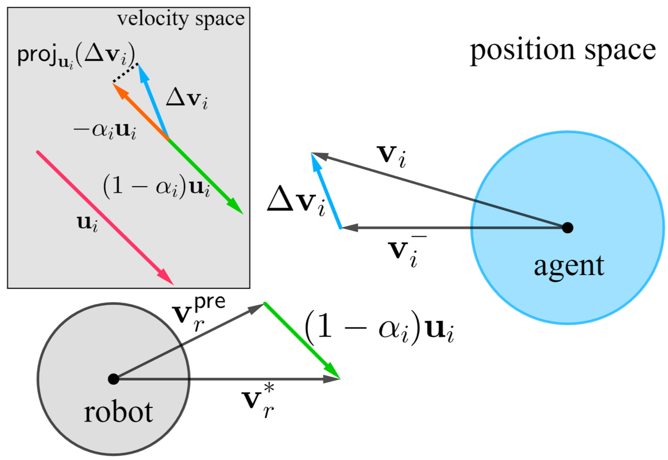


Fig. 5. Geometry behind the projection estimator for e_i.

The above-mentioned equation leads to the second order equation

$$\beta_1\tau_i^2 + \beta_2\tau_i + \beta_3 = 0 \tag{14}$$

with $\beta_1 = (\tilde{v}_i^x)^2 + (\tilde{v}_i^y)^2, \beta_2 = 2\tilde{v}_i^x(p_r^x - p_i^x) + 2\tilde{v}_i^y(p_r^y - p_i^y)$, and $\beta_3 = R^2 - (p_r^x - p_i^x)^2 - (p_r^y - p_i^y)^2$. The solutions of the quadratic equation can be the following.

1) If $\beta_2^2 < 4\beta_1\beta_3$, then there is no intersection and, therefore, no potential collision. As a consequence, we set $\tau_i = \infty$.
2) If $\beta_2^2 = 4\beta_1\beta_3$, then the solution is given by a double root $\tau_i = \frac{-\beta_2}{2\beta_1}$. Otherwise, if $\beta_2^2 > 4\beta_1\beta_3$, then there are two real roots. In both cases, the following reasoning holds.
   - If both roots are positive, then we take the minimum among them.
   - If both roots are negative, then there is no collision and $\tau_i = \infty$.
   - If one root is positive and one root is negative, then collision has already happened and, therefore, motion is terminated.

Note that τ_i is inexpensive to compute and enables time-to-collision checking in real-time.

The next step is to define e_i and develop a method to compute it from sensor observations. According to the definition of the nonlinear opinion dynamics in (10), e_i represents the opinion that agent i has on the value of the shifted degree of cooperation o_i. Recalling the reasoning behind the nonlinear opinion dynamics design in (5), the consensus term $a_io_i + c_ie_i$ aims at fusing the values that the robot and agent i have on the degree of cooperation o_i. In the case of the robot, this is simply the current value of o_i; in the case of agent i, this is precisely the opinion that agent i has on the value of the shifted degree of cooperation o_i. In typical opinion dynamics formulations it is assumed that these values can be communicated. This is not the case in this work, so we need to recover e_i using an estimation method that relies solely on onboard sensing.

Fig. 5 depicts the geometry of the problem. At a certain instant, the robot has a predefined velocity $\mathbf{v}_r^{\mathrm{pre}}$ but executes $\mathbf{v}_r^*$ to avoid collision. The difference between both velocities is $\mathbf{v}_r^* - \mathbf{v}_r^{\mathrm{pre}} = (1-\alpha_i)\mathbf{u}_i$ due to constraint (8b) in (8). On the other hand, agent i executes $\mathbf{v}_i$, and executed $\mathbf{v}_i^-$ in the previous instant, where $\mathbf{v}_i^- \in \mathbb{R}^2$ denotes the velocity of agent i sensed by the robot at the previous time instant. Since the robot has been in charge of exerting $(1-\alpha_i)\mathbf{u}_i$, the remaining part of $\mathbf{u}_i$ to avoid collision must be exerted by agent i, i.e., $\alpha_i\mathbf{u}_i$. Recall that $\mathbf{u}_i$ represents a change in the velocity. Henceforth, the remaining part of $\alpha_i\mathbf{u}_i$ is exerted through the change in velocity of agent i, namely, $\Delta\mathbf{v}_i = \mathbf{v}_i - \mathbf{v}_i^-$. However, in general, $\Delta\mathbf{v}_i \cdot \alpha_i\mathbf{u}_i \neq 0$, i.e., vectors $\Delta\mathbf{v}_i$ and $\alpha_i\mathbf{u}_i$ are not aligned. This means that it is the component of $\Delta\mathbf{v}_i$ parallel to $\alpha_i\mathbf{u}_i$ the one that exerts the change in velocities to avoid collision. Therefore, if we compute the magnitude of vector component of $\Delta\mathbf{v}_i$ parallel to $\mathbf{u}_i$ and compare it to the magnitude of $\mathbf{u}_i$, then we can obtain α_i.

These insights motivate the following projection estimator of e_i:

$$e_i = \mathrm{tanh}\left(\varepsilon\left(\frac{||\mathrm{proj}(\Delta\mathbf{v}_i, \mathbf{u}_i)||}{||\mathbf{u}_i||} - \frac{1}{2}\right)\right). \tag{15}$$

The projection operator $\mathrm{proj}(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a}\cdot\mathbf{b}}{\mathbf{b}\cdot\mathbf{b}}\mathbf{b}$ projects the vector $\Delta\mathbf{v}_i$ on vector $\mathbf{u}_i$. Furthermore, by normalizing with the norm $||\mathbf{u}_i||$, the estimate of α_i is extracted. The other operations translate the estimated degree of cooperation to the estimated shifted degree of cooperation e_i, where $\varepsilon > 0$ tunes how sensitive is the tanh function to changes in $\Delta\mathbf{v}_i$. In this sense, we remark that (15) is not a perfect estimator since it only relies on the sensor observations of the change in velocities $\Delta\mathbf{v}_i$. This motivates the use of the function tanh, which simultaneously restricts the domain of the estimation to $[-1, 1]$ and allows a fast evolution of the estimated shifted degree of cooperation. For instance, if the robot observes that agent i moves straight to it and $\Delta\mathbf{v}_i \approx 0$, then this means that $e_i \approx -1$ and the robot estimates that agent i is not cooperating. It is worth noting that the projection estimator in (15) directly uses the velocity measurements from onboard sensors, which are subject to disturbances. However, designing the adaptive law as a nonlinear opinion dynamic naturally provides robustness against small perturbations, and exploits past information in the opinion estimation by means of the memory term $-d_io_i$ [73]. Therefore, the proposed adaptive law can be framed as a nonlinear estimator with a dynamic model (10) and a measurement model (15). The sensitivity and robustness of the general formulation of the nonlinear opinion dynamics model is analyzed in previous works [72]. The next section analyzes specific considerations on how to tune the parameters of AVOCADO.

C. Nonlinear Opinion Dynamics Adaptive Law Tuning

To achieve fast and effective adaptation to the unknown degree of cooperation of the agent, the nonlinear opinion dynamics adaptive law of AVOCADO must be appropriately parameterized.

First, we examine the stability properties of the adaptive law and the conditions under which bifurcation, i.e., instability of the initially stable equilibrium point of (10), happens. To do so, we use the expression $\mathrm{sech}(\bullet) = 1/\cosh(\bullet)$.

Proposition 1: Let $d_i > 0$, $a_i > 0$ and $b_i \in [-1, 1]$. Then, $o_i = b_i/d_i$ is an unstable equilibrium point of the nonlinear opinion dynamics adaptive law in (10) if $A_i > 1/(a_i \text{sech}^2(a_i \frac{b_i}{d_i}))$.

Proof: Initially, $A_i = 0$, $e_i = 0$ and $\dot{o}_i = 0$. Therefore, the robot has an initial opinion $o_i = b_i/d_i$ that is an equilibrium of the nonlinear opinion dynamics adaptive law in (10). Nevertheless, whether this equilibrium is stable or unstable depends on the parameters of the adaptive law. Let

$$\lambda = -d_i + d_i A_i a_i \text{sech}^2\left(a_i \frac{b_i}{d_i}\right) \tag{16}$$

be the unique eigenvalue of (10) after linearization at $o_i = b_i/d_i$. Equilibrium becomes unstable when $\lambda > 0$. Assuming that a_i, d_i and b_i are fixed, and taking into account that $\text{sech}(\bullet) \in (0, 1]$ in all its domain, $o_i = b_i/d_i$ becomes an unstable equilibrium when $A_i > 1/(a_i \text{sech}^2(a_i \frac{b_i}{d_i}))$. □

Corollary 1: If $b_i = 0$, then $\text{sech}^2(a_i \frac{b_i}{d_i}) = 1$ and the condition on the attention level reduces to $A_i > 1/a_i$. Since $\text{sech}^2(a_i \frac{b_i}{d_i}) \in (0, 1)$, $A_i > 1/a_i$ is an upper bound on the attention value beyond which the equilibrium $o_i = b_i/d_i$ becomes an unstable equilibrium and, therefore, $A_i > 1/a_i$ is a sufficient condition for $o_i = b_i/d_i$ being an unstable equilibrium.

When the robot and agent i are far from each other, the dynamics of the attention in (11) are designed such that $A_i = 0$. When the robot and agent i approach each other in a trajectory of potential collisions, $\mathsf{tanh}(\kappa_i \tau_i^{-1}) > 0$ and attention evolves toward $A_i > 0$, with a speed determined by δ_i. Since the attention dynamics in (11) is a linear system with bounded input, the attention is stable with equilibrium given by $A_i^{\mathsf{eq}} = \mathsf{tanh}(\kappa_i (\tau_i^{\mathsf{eq}})^{-1})$. Therefore, κ_i must be large enough to ensure that $A_i > \frac{1}{a_i}$ when the time to collision τ_i is large enough to avoid collision. Moreover, since $A_i \in [0, 1]$, a_i must be designed such that $\frac{1}{a_i} < 1$ to ensure that the condition $A_i > \frac{1}{a_i}$ is possible for some τ_i.

Now, we examine the stability properties of the nonlinear opinion dynamics adaptive law in (10) after bifurcation.

Proposition 2: Let, $d_i > 0$, $a_i > 0$, and $b_i \in [-1, 1]$. Assume that $A_i > \frac{1}{a_i}$ holds. Then, the nonlinear opinion dynamics adaptive law in (10) exhibits two stable equilibrium points, given by the following expression:

$$o_i^{\mathsf{eq}} = A_i \mathsf{tanh}(a_i o_i^{\mathsf{eq}} + c_i e_i) + \frac{b_i}{d_i}. \tag{17}$$

Besides, $|o_i^{\mathsf{eq}}| \in [\frac{|b_i|}{d_i} - 1, \frac{|b_i|}{d_i} + 1]$.

Proof: From the assumption that $A_i > \frac{1}{a_i}$ holds, in the transition $A_i = \frac{1}{a_i}$ bifurcation happens [72]. The dynamics in (10) are bounded due to the term $-d_i o_i$, so equilibrium(s) point(s) exist(s) after bifurcation. By enforcing the equilibrium condition $\dot{o}_i = 0$ in (10), (17) is obtained. We recall that $A_i \in [0, 1]$ and $|\mathsf{tanh}(\bullet)| \in [0, 1]$. Therefore, the absolute value of the equilibrium $|o_i^{\mathsf{eq}}|$ is bounded and such that $|o_i^{\mathsf{eq}}| \in [\frac{|b_i|}{d_i} - 1, \frac{|b_i|}{d_i} + 1]$. □

If there is no bias, $b_i = 0$ and $o_i^{\mathsf{eq}} \in [-1, 1]$; otherwise, the bias introduces a shift in the domain that naturally encodes a priori knowledge on the degree of cooperation of the agent. Proposition 2 motivates the use of the scaling factor d_i in the attention A_i: in the absence of this factor, the expression of the equilibrium points is $o_i^{\mathsf{eq}} = \frac{A_i \mathsf{tanh}(a_i o_i^{\mathsf{eq}} + c_i e_i) + b_i}{d_i}$, leading to an asymmetrical domain for $|o_i^{\mathsf{eq}}|$ with respect to b_i/d_i, i.e., not symmetrical with respect to the initial equilibrium before bifurcation. This asymmetry would derive in an intrinsic bias toward any of the two subsequent equilibrium points, which is an undesired effect that we overcome by including the scaling factor d_i.

Finally, c_i shall be designed to guarantee that the nonlinear term $\mathsf{tanh}(\bullet)$ in the adaptive law is enough sensitive to the behavior of agent i, reacting fast enough to avoid collisions. On the other hand, ε in (15) tunes how sensitive is the estimator of e_i to changes in the speed of agent i.

In summary, the nonlinear opinion dynamics adaptive law depends on six parameters as follows.

1) $\varepsilon > 0$ determines how sensitive is the estimator of e_i and should be chosen large enough to ensure $e_i \approx 1$ with enough time to avoid collision but small enough to avoid noisy estimations.
2) $\kappa_i > 0$ and $\delta_i \in [0, 1)$ determines how fast the adaptive law approaches the bifurcation point, and therefore establishes when the robot decides its opinion about the degree of cooperation of agent i.
3) $d_i > 0$ sets the convergence speed of the dynamics of the adaptive law, associated to the forgetting factor $-d_i o_i$.
4) a_i determines the attention level necessary to reach the bifurcation, taking into account the value of d_i.
5) b_i is a bias that encodes potential prior knowledge on the (shifted) degree of cooperation of agent i.
6) c_i weights the importance of the estimate e_i compared to the current value of $a_i o_i$ such that, if $a_i >> c_i$, the robot tends to preserve its opinion over the (shifted) degree of cooperation, or heavily relies on the estimator if $c_i >> a_i$.

D. Exploiting Attention to Avoid Symmetry Deadlocks

The previous sections address the problem of unknown degree of cooperation in collision avoidance. However, there is an additional issue, typical in VO-based methods, such as ORCA, that is characterized by navigation deadlocks under symmetrical configurations [86]. Among other mechanisms, VO-based methods usually solve symmetry deadlocks by injecting some arbitrary noise in the perceived velocity, where this amount is typically handcrafted. Instead, we propose an adaptive noise injection mechanism that exploits the attention A_i of the opinion dynamics to correlate the amount of noise with safety.

Specifically, our solution consists in injecting some small noise to the perceived velocity of agent i when the agent is far from the robot, decreasing the noise to zero when the robot and the agent are close to a potential collision. When the robot and the agent are far from each other, the impact of noise is negligible because attention $A_i \approx 0$ and bifurcation does not happen. This preserves the neutrality of the opinion on the (shifted) degree of cooperation. When the robot and the agent are close to collision, noise is removed, so the performance guarantees on the adaptive law and the collision avoidance are preserved.

Mathematically speaking, we reformulate (6) as

$$\mathbf{u}_i = \underset{\mathbf{v}_r \in \partial \mathsf{VO}_i}{\arg\min} ||\mathbf{v}_r - (\mathbf{v}_r^{\mathsf{pre}} - \mathbf{v}_i^{\mu})|| - (\mathbf{v}_r^{\mathsf{pre}} - \mathbf{v}_i^{\mu}) \quad (18)$$

where

$$\mathbf{v}_i^{\mu} = \mathbf{v}_i + (1 - A_i)\mu(\sigma) \quad (19)$$

with $\mu(\sigma) \sim \mathcal{U}(-\sigma, \sigma)$ a uniformly distributed perturbation bounded by $\sigma > 0$.

In essence, our method adds a uniformly distributed noise to the velocity of agent i sensed by the robot. This noise depends on the attention mechanism developed for the nonlinear opinion dynamics adaptive law. This perturbed velocity is the one used to compute vector $\mathbf{u}_i$ and the OCA_i^{τ} constraint sets. Hence, the noise perturbs the potential symmetry between $\mathbf{v}_r^{\mathsf{pre}}$ and $\mathbf{v}_i$ only when the robot and agent i are sufficiently close to each other. The intensity of the perturbation increases when the distance between robot and agent i decreases in order to ensure that there is an effective deadlock breaking when robot and agent i approach each other. It is interesting to remark that this deadlock breaking approach can also be applied to other VO-based methods such as ORCA, since it only requires the implementation of the attention mechanism in (11).

AVOCADO is summarized in Algorithm 1. By combining a geometrical approach with a nonlinear opinion dynamics adaptive law, AVOCADO is able to adapt in real-time to the unknown degree of cooperation of the agents in a multiagent setting. The unknown degree of cooperation of the agent induces an additional layer of complexity in collision avoidance, since, as it is shown in Section V, an incorrect assumption on the degree of cooperation leads to collision to existing geometrical, learning and model predictive approaches. On the other hand, AVOCADO is agnostic to the nature of the agents the robot encounters, so the robot can take a fast and flexible decision irrespective of the dynamic model of the agent. AVOCADO explicitly couples prediction with planning by estimating the effect its motion is producing in the other agents it encounters through the degree of cooperation α_i $\forall i \in \mathcal{N}$. In other words, the degree of cooperation α_i, unless 0, expresses the fact that the motion of agent i is influenced by the velocity exerted by the robot. By designing an adaptive law that estimates α_i in the continuous domain $\alpha_i \in [0, 1]$, our method is able to deconflict the undesirable effects of such coupling by predicting how the agent reacts to the robot motion and plan the collision avoidance maneuver accordingly. No communication is involved, only relying on the perceived position and velocity of the nearby agents. All the mathematical operations are inexpensive to compute, except for the optimization program in line 11. Nonetheless, since it entails a linear program, with an efficient implementation it is proven that a robot can process thousands of nearby agents in milliseconds [12]. We validate the computational simplicity of AVOCADO in Section V.

V. Simulated Results

First, we evaluate AVOCADO in simulated scenarios. In Section VI, we describe the results obtained in real settings with robots and humans.

Algorithm 1: AVOCADO.

1: **Parameters**: $r_s, r_p > 0$, $\varepsilon > 0$, $\kappa_i > 0$, $d_i > 0$, $a_i, b_i, c_i \in \mathbb{R}$, $\sigma > 0$, $\delta_i \in [0, 1)$
2: **for** all t **do**
3: Get $\mathbf{p}_r$ and $\mathbf{v}_r^{\mathsf{pre}}$ from a higher-level planner.
4: Measure $\mathbf{p}_i, \mathbf{v}_i$ $\forall i \in \mathcal{N}$ from sensors and get the stored $\mathbf{v}_i^-$ $\forall i \in \mathcal{N}$.
5: Update the attention level A_i using $\mathbf{p}_r$, $\mathbf{v}_r^{\mathsf{pre}}$, $\mathbf{p}_i$ and $\mathbf{v}_i$ through (11) and (14), for all $i \in \mathcal{N}$.
6: Apply (19) to compute $\mathbf{v}_i^{\mu}$ $\forall i \in \mathcal{N}$.
7: Calculate $\mathbf{u}_i$ through (18).
8: Estimate e_i through (15).
9: Update the opinion on the shifted degree of cooperation using the adaptive law in (10).
10: Build all admissible sets OCA_i through (7), for all $i \in \mathcal{N}$.
11: Solve optimization problem (8) to obtain $\mathbf{v}_r^*$.
12: Apply $\mathbf{v}_r^*$ and store $\mathbf{v}_i$ as $\mathbf{v}_i^-$ $\forall i \in \mathcal{N}$.
13: **end for**

In Section V-A we analyze the impact of the different parameters in the behavior of AVOCADO. Next, in Section V-B we conduct extensive multiagent simulations using two navigation settings. For each setting, we compare qualitatively and quantitatively AVOCADO with existing state-of-the-art planners against other cooperative robots and noncooperative agents. In all the simulated scenarios, we consider that robots are cooperative, i.e., they act using the same motion planner under comparison. For instance, if AVOCADO is assessed, then all robots use AVOCADO. Importantly, in our article, a robot being cooperative does not mean knowledge on the degree of cooperation nor communication exchange, but just the robot uses a planner to avoid collisions. In contrast, agents are noncooperative, i.e., they can not perceive the robots. Noncooperative agents resolve (8), but their neighbor set $\mathcal{N}_i$ never includes the robots, only other agents. This is done to ensure collision avoidance among agents but complete noncooperation with the robots, so the behavior is more complex than a simple dynamic obstacle with a fixed trajectory. A mixed cooperative/noncooperative navigation scenario involves cooperative robots and noncooperative agents. Finally, robots and agents have a disc shape of radius $r_s = 0.2$m and a sensor range of $r_p = 2.5$ m.

The maximum velocity, $\mathbf{v}_r^{\mathsf{max}}$, of the robots is set to 1m/s and $\mathbf{v}_i^{\mathsf{max}}$ of the agents is set to 0.75 m/s. The agents have a lower velocity than the robots, allowing the robots to escape if the agents move quickly toward them. For robots using AVOCADO, $\mathbf{v}_r^{\mathsf{pre}} = \mathbf{v}_r^{\mathsf{max}} \frac{\mathbf{p}_{r,j}^* - \mathbf{p}_{r,j}^t}{||\mathbf{p}_{r,j}^* - \mathbf{p}_{r,j}^t||}$, where $\mathbf{p}_{r,j}^*$ denotes the desired goal for robot j and $\mathbf{p}_{r,j}^t$ is the position of robot j at time t. Therefore, robots using AVOCADO are not helped by any higher level planner to choose the desired velocity.

A. Head-On Scenarios

We simulate head-on scenarios to analyze the impact of the different parameters in AVOCADO. A robot and either a

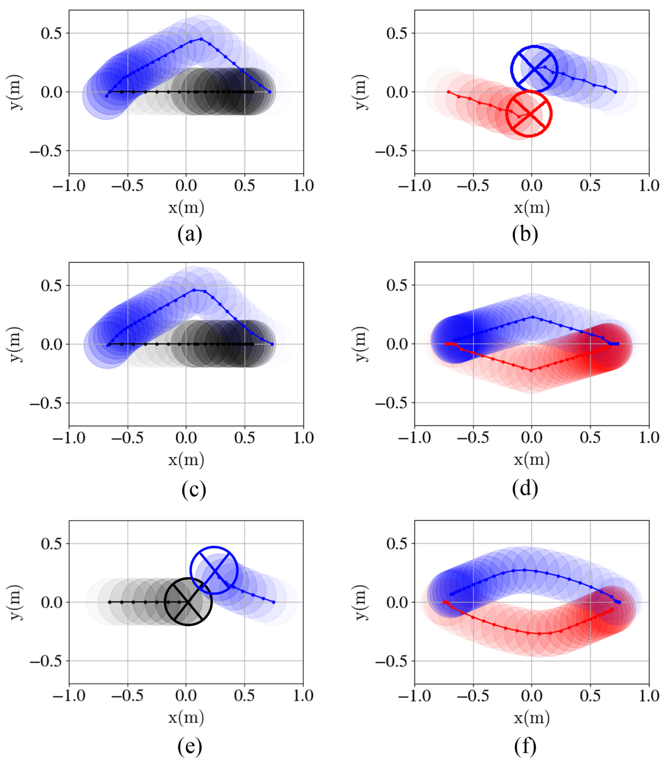


Fig. 6. Head-on scenarios for different ratio b_i/d_i and degree of cooperation. The robot (blue) navigates using AVOCADO, whereas the agent is either noncooperative (black, left column) or navigates using AVOCADO (red, right column). The motion of the robot and the agent is represented with disks of increasing transparency as time advances, and solid dots represent the center of the disks. Collision is marked with crosses surrounded by circles. (a) $b_i/d_i = -0.9$. (b) $b_i/d_i = -0.9$. (c) $b_i/d_i = 0$. (d) $b_i/d_i = 0$. (e) $b_i/d_i = 0.9$. (f) $b_i/d_i = 0.9$.

TABLE I
DEFAULT PARAMETER VALUES

Parameter	a_i	b_i	d_i	κ_i	ε	δ_i	b_i
Value	0.3	0.7	2	14.15	3.22	0.57	0

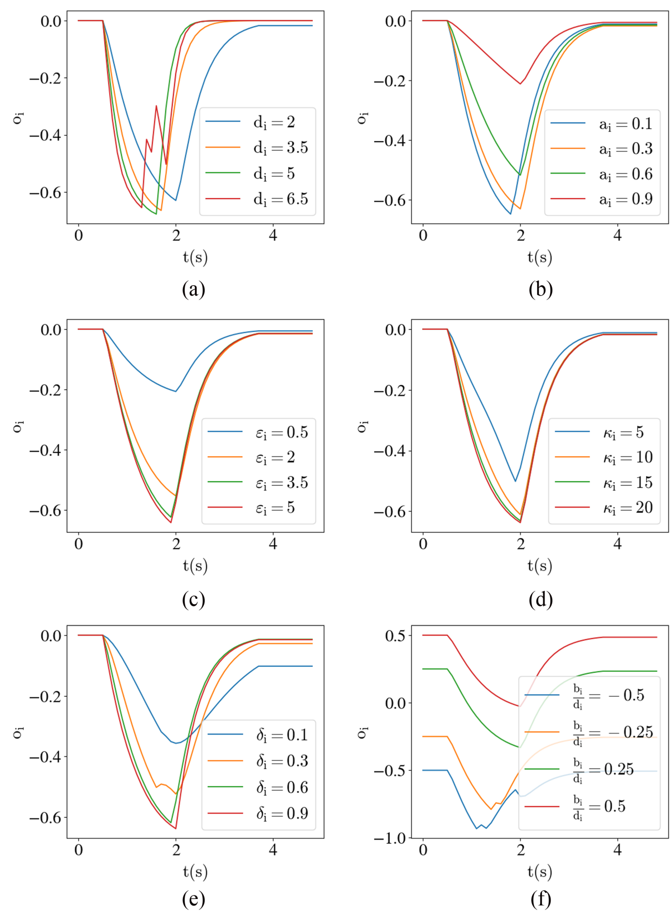


Fig. 7. Evolution of the degree of cooperation in the head-on scenario against a noncooperative agent, varying the value of each parameter of AVOCADO. (a) Different d_i. (b) Different a_i, $c_i = 1 - a_i$. (c) Different ε. (d) Different κ_i. (e) Different δ_i. (f) Different b_i.

cooperative robot or a noncooperative agent face each other, where their goals are the starting position of the other. Fig. 6 illustrates these scenarios for different values of bias b_i (a, c, e for the noncooperative agent; b, d, f for the cooperative robot). The robot using AVOCADO starts, in scenarios a) and b), assuming that the other robot or agent is noncooperative with a low degree of cooperation; in scenarios c) and d), the robot initially assumes reciprocity, with a degree of cooperation equal to 0.5; and in e) and f) the robot initially assumes a great degree of cooperation. Extreme bias values can lead to collisions, as in b) and e). In a), the robot effectively avoids the collision with the agent, as the highly negative biased initial degree of cooperation (pessimistic robot) corresponds to reality. However, when the bias is incorrect, even if the agent is cooperative as in b), the robot takes some time to recover from the incorrect prior, resulting in collision. The opposite conflict arises when there is an initial highly positive bias in the degree of cooperation (optimistic). In case e), the robot is unable to correct the estimated degree of cooperation fast enough to realize that the agent is noncooperative and take all the responsibility to avoid collision. Collision is avoided in case f), but the extreme bias leads to sub-optimal trajectories. Finally, a balanced bias allows the robot to react effectively to noncooperative agents in c) and cooperative agents in d).

The impact of the other parameters is evaluated by studying the evolution of o_i when the parameters change. The default parameters are specified in Table I. Now, we focus on the noncooperative head-on scenario.

Fig. 7 depicts the evolution of o_i for the different cases. As the robot detects that agent i does not cooperate, o_i decreases, trying to reach $o_i = -1$. Finally, when the collision is avoided and $A_i = 0$, o_i recovers to the initial value. In some cases, as in Fig. 7(e) with $\delta = 0.1$, o_i does not recover to the initial value because agent i leaves the sensor range of the robot before letting the nonlinear opinion dynamics to converge to $o_i = b_i/d_i$. First, regarding d_i (10), greater values imply a greater forgetting factor and therefore, a faster convergence. For very large values ($d_i = 6.5$), the nonlinear opinion dynamics are too reactive to the estimate e_i and to the noise introduced (19), leading to abrupt changes in o_i. Second, regarding a_i, c_i (10), since $c_i = 1 - a_i$ and $a_i \in \{0.1, 0.3, 0.6, 0.9\}$, greater values of a_i imply less

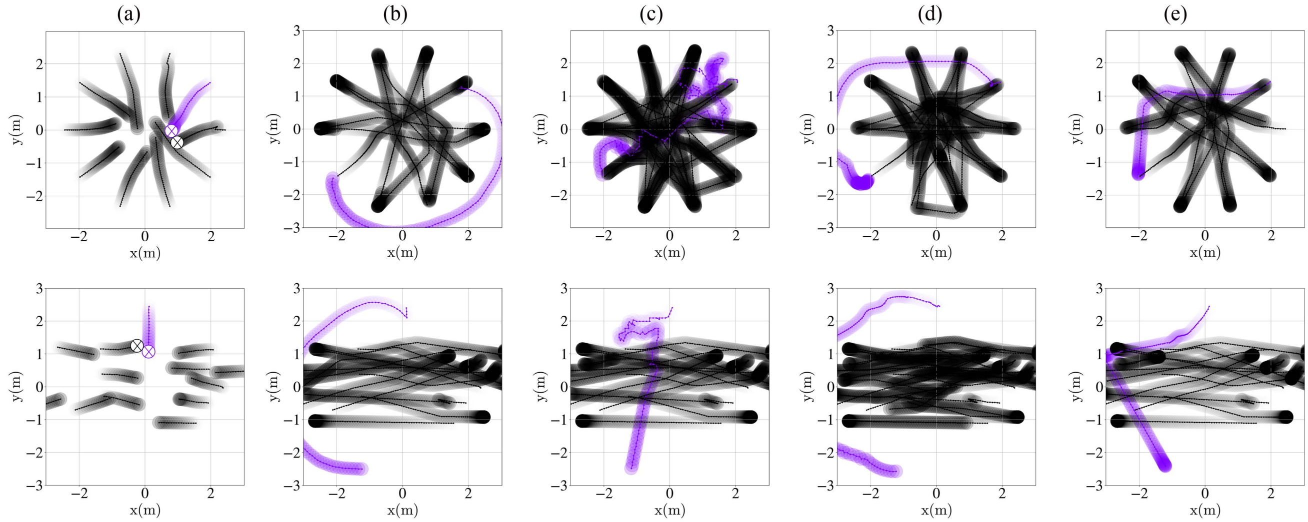


Fig. 8. Noncooperative *circle* (top) and *crossing* (bottom) scenarios with ten agents and different planners. The robot has a purple color, and the noncooperative agents are in black. When the robot collides with an agent, it becomes transparent. The episode finishes when all the agents and the robot reach the goal or collide, or the simulation lasts more than 100 s. (a) ORCA. (b) T-MPC. (c) SARL. (d) RVO-RL. (e) AVOCADO.

reactivity against changes in the estimate e_i and, henceforth, slowest convergence of the nonlinear opinion dynamics. Third, regarding ε (15), it is observed that beyond some value around $\varepsilon = 1$, the estimate of e_i is reactive enough to guarantee a fast convergence of the nonlinear opinion dynamics. Fourth, the behavior of κ_i (11) is similar to that of ε, in the sense that there is a point beyond which increasing κ does not provide any improvement. Fifth, the same holds for δ_i (11). Finally, regarding the ratio b_i/d_i (10), it is very interesting to observe that different values lead to different initial and final equilibrium, equal to $o_i = b_i/d_i$. In all the cases, since the agent is always noncooperative, o_i evolves toward a low value in order to take the responsibility of avoiding the collision. The plots of Fig. 7 show that, in general, very low parameters' values are not desirable, as they make the opinion grow slowly and the robot to react late. Nevertheless, very large values can make the nonlinear opinion dynamics adaptive law to be too sensitive to changes in the agent and the estimate e_i, leading to a nonsmooth evolution of o_i. Therefore, it is suggested to design the parameters as low as possible to guarantee collision avoidance but obtain fluid maneuvers. The nominal parameters described in Table I have been tuned using Bayesian Optimization, constrained the domain of the parameters according to the conclusions drawn from Fig. 7.

B. Multiagent Scenarios

After studying the influence of the different parameters in AVOCADO, we conduct a series of multiagent simulations to evaluate the performance of AVOCADO compared to existing state-of-the-art planners. We compare AVOCADO with ORCA [12] and RVO-RL [51] as baselines planners that consider reciprocal collision avoidance, and T-MPC [23], [68] and SARL [87] as baseline planners for noncooperative collision avoidance. The former is based on an MPC formulation and the latter is a neural-network-based planner trained using reinforcement learning. In this sense, SARL is retrained to satisfy the constraints in perception from the limited sensing radius of robots and agents. Moreover, we set $\sigma = 0.0001$ for AVOCADO.

We design two evaluation settings. The *circle* scenario is characterized by an evenly spaced initial position of robots and agents, forming a circle and with the goal of reaching the initial position of the opposite robot or agent. On the other hand, the *crossing* scenario is characterized by a random initial position of all the agents and robots in the border of a square. Robots and agents must navigate to a individually assigned random goal in the opposite side of the square. Cooperative robots are placed in the sides oriented to one axis and noncooperative agents are placed in the sides oriented to the other axis. This arrangement enforces that cooperative robots traverse noncooperative agents that move perpendicularly as a traffic flow they have to cross. Besides, to ensure that robots do not take trivial motion strategies (e.g., wait until all the agents have reached the goal), once an agent reaches its goal, the goal is modified to be the initial position of the agent, so the environment is always dynamic.

We show the qualitative behavior of the robots in noncooperative *circle* and *crossing* scenarios in Fig. 8. The planners address the navigation conflict at the center of the circle in two different ways. T-MPC and RVO-RL completely avoid dangerous zones by taking a detour; this might not be possible in a constrained scenario with boundaries. Among the other three approaches, ORCA is the one that takes more risks, since it assumes reciprocity in the degree of cooperation, colliding in both scenarios. SARL and AVOCADO achieve adaptation to the unknown degree of cooperation of the agents and other robots and manage to reach the goal in both scenarios; nonetheless, the movements exerted by the robots using AVOCADO are much more seamless than those exerted by SARL, which is a key aspect when transferring navigation policies to real robots.

Fig. 9 shows examples of scenarios only populated with robots. Fig. 9(a) demonstrates the deadlocks suffered by reciprocal approaches, such as ORCA, when symmetries are

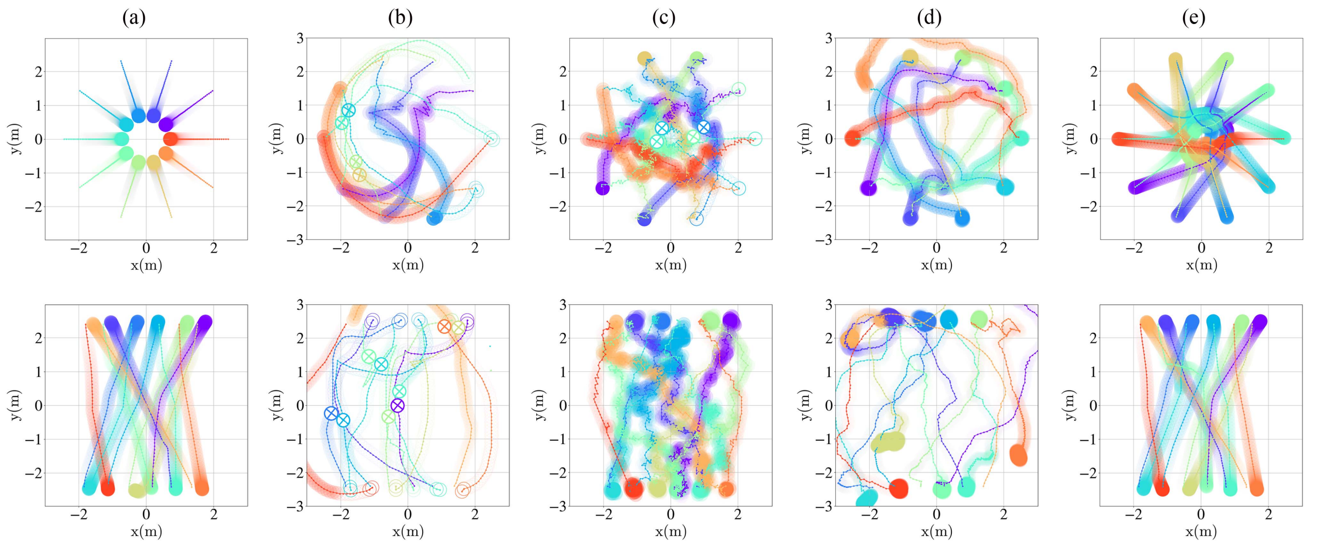


Fig. 9. Cooperative *circle* (top) and *crossing* (bottom) scenarios with ten robots and different planners. Each robot is depicted with a different color. The representation follows the same rules in Fig. 8. Collisions are marked with crosses surrounded by circles. (a) ORCA. (b) T-MPC. (c) SARL. (d) RVO-RL. (e) AVOCADO.

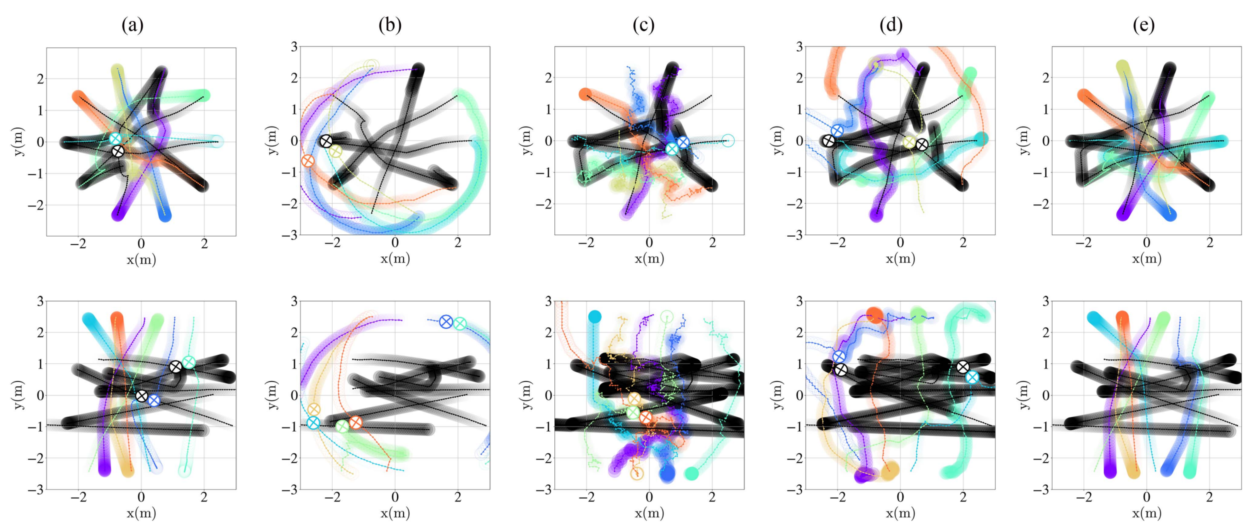


Fig. 10. Mixed cooperative/noncooperative *circle* (top) and *crossing* (bottom) scenario with four agents, six robots, and different planners. Each robot is depicted with a different color. The representation follows the same rules in Fig. 8. Collisions are marked with crosses surrounded by circles. (a) ORCA. (b) T-MPC. (c) SARL. (d) RVO-RL. (e) AVOCADO.

encountered. AVOCADO overcomes this problem, as seen in Fig. 9(e), by exploiting the attention mechanism in the nonlinear opinion dynamics adaptive law. T-MPC and RVO-RL take long unnecessary detours, as they are designed for scenarios with fewer robots and to take larger safety margins. T-MPC even presents many collisions, as it is not prepared to interact with other robots, and some of the RVO-RL robots do not reach the goal before the time out of 100 s. Trajectories of robots using SARL are, again, very irregular, probably due to a reciprocal dance problem, especially noticeable in *circle* scenarios where this leads to collisions. All the robots using AVOCADO reach their goals avoiding collisions, with short path lengths and fluid trajectories, in both scenarios.

Mixed cooperative/noncooperative scenarios are the most challenging for most of the planners, as they assume some sort of degree of cooperation, either reciprocity or complete absence of cooperation. Fig. 10 provides examples of such scenarios. T-MPC and RVO-RL avoid passing through the center of the stage, as there are many noncooperative agents. However, they encounter cooperative robots during the detour, so they are unable to avoid collisions. SARL faces again the reciprocal dance problem when evading both noncooperative agents and other robots avoiding collisions. ORCA and AVOCADO perform similar trajectories, but AVOCADO adaptation capabilities makes it safer, avoiding all collisions unlike ORCA.

We conduct a systematic series of *circle* and *crossing* runs to extract quantitative metrics to compare all the planners, using an increasing total number of agents. The circle size of *circle* scenarios is set to $\max(2.5, \frac{2.3\mathrm{N}r_s}{\pi})$m and the square size of *crossing* scenarios is set to $1.5\mathrm{N}r_s$m. These numbers guarantee that there is enough space for the initial and final position of all robots and agents while enforcing collision conflicts. Simulation runs range from $\mathsf{N} = 10$ to $\mathsf{N} = 25$ in steps of 3. For each value of N, we test five different proportions P of cooperative robots. In particular, we set the number of cooperative robots equal to $\lceil \mathsf{PN} \rceil$, with $\mathsf{P} = \{0.01, 0.25, 0.5, 0.75, 1\}$, where $\lceil \bullet \rceil$ is the operator that rounds a real number to the closest greater integer. The others are noncooperative agents. We run 128 random scenarios for each combination of planner, number of cooperative robots and noncooperative agents. Each run in *circle* scenarios randomizes the identity of each agent (noncooperative or the planner under evaluation), while each run in *crossing* scenarios randomizes the initial and goal positions of each agent. We compare the state-of-the-art planners with four versions of AVOCADO with the following parameters.

1) AVOCADO_1: Table I.
2) AVOCADO_2: Table I but $d_i = 5$.
3) AVOCADO_3: Table I but $b_i = 1$ ($\frac{b_i}{d_i} = 0.5$).
4) AVOCADO_4: Table I but $b_i = -1$ ($\frac{b_i}{d_i} = -0.5$).

We define the success rate as a evaluation metric. Let M be the number of runs; $\text{succ}_{j,k} \in \{0, 1\}$ is the indicator that is equal to 1 if $||\mathbf{p}_{r,j} - \mathbf{p}^*_{r,j}|| < \xi$ at the end of run k or 0 otherwise (it is also zero when the robot j at run k collides), where $\mathbf{p}_{r,j}$ is the position of robot j, $\mathbf{p}^*_{r,j}$ is the desired goal of robot j, and $\xi > 0$ is a small tolerance. Then, the success rate is defined as

$$\frac{1}{\mathsf{MN}} \sum_{k=1}^{\mathsf{M}} \sum_{j=1}^{\mathsf{N}} \text{succ}_{j,k}.$$

The success rates are in Figs. 11 and 12 for *circle* and *crossing* scenarios, respectively. AVOCADO outperforms all other approaches in success rate. AVOCADO_2, with a higher value of d_i than AVOCADO_1, has a greater success rate in *circle* scenarios where the number of cooperative robots is greater, as it is more sensitive to the introduced noise that breaks symmetries. AVOCADO_1 is more stable and has a better performance in partially cooperative scenarios. AVOCADO_3 presents a worse success rate due to the bias, that assumes low degrees of cooperation when it is often not true. In this sense, it is very interesting to see how AVOCADO_4, with a bias toward great degrees of cooperation, presents the worst results among the AVOCADO versions in Fig. 11 unless in fully cooperative cases. However, AVOCADO_4 achieves the best success rates in Fig. 12. This is probably due to the fact that, in *crossing* scenarios, the robot first faces noncooperative perpendicular traffic, where being cautious is desirable; meanwhile, in *circle* scenarios the robot faces agents coming in their same direction, leading to conflicts similar to the one previously seen in Fig. 6(b). ORCA experiences deadlocks from geometrical symmetries, so it fails in all cooperative *circle* scenarios. The performance of T-MPC degrades when the number of cooperative agents increases, since, as it is observed in [23] and [68], the method is suited for scenarios with less than ten robots or agents. RVO-RL, tuned for fully cooperative environments, achieves its best performance for $\mathsf{P} = 1$, decreasing the success rate as noncooperative agents

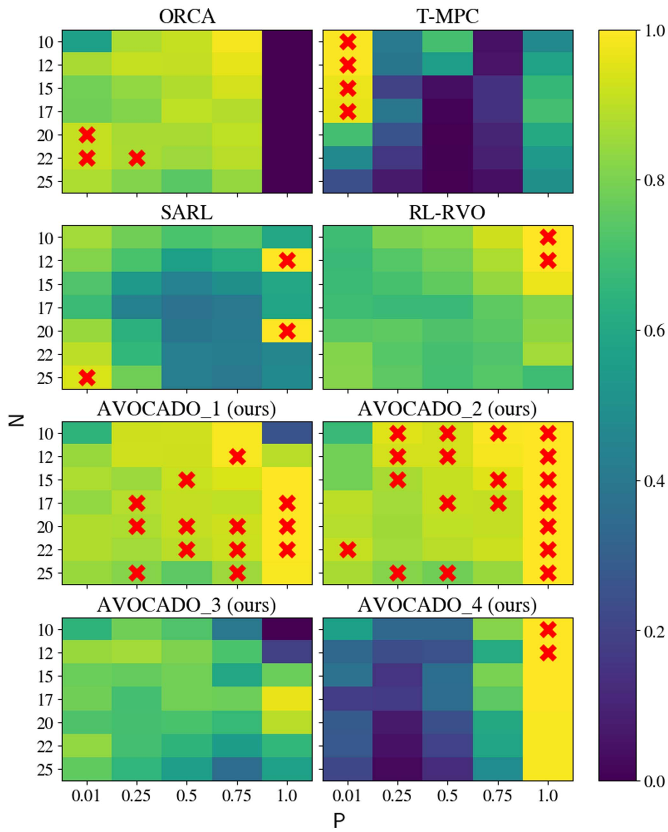


Fig. 11. Success rates of robots with the different planners in *circle* scenarios. Red crosses mark the best planner. The color bar refers to success rate as defined in Section V-B, whereas P refers to the proportion of cooperative robots. The yellower the better.

appear. SARL, due to the nonfluid behavior, obtains poor success rates.

We show in Fig. 13 the mean navigation time that every planner takes to reach the goal in *square* scenarios with $\mathsf{P} = 0.5$. The mean time to goal is computed taking into account only the robots that do not collide

$$\frac{1}{\mathsf{MN}} \sum_{k=1}^{\mathsf{M}} \sum_{\substack{j=1 \\ \text{succ}_{j,k}=1}}^{\mathsf{N}} t_{j,k}$$

where $t_{j,k} > 0$ denotes the instant when robot j reaches its goal at run k. The results are aligned with the qualitative results depicted in Figs. 8–10. RL-RVO and SARL exhibit the largest time to goal because their trajectories are irregular and present many detours. Meanwhile, the rest of the planners reach the goal in similar times. T-MPC manifests slightly lower times to reach the goal even tough the resulting trajectories are longer than those of AVOCADO or ORCA. This is due to the fact that T-MPC exerts higher velocities and it is, therefore, more risky, which explains its success rate metrics shown in Figs. 11 and 12.

Finally, regarding computation times, Fig. 14 collects the mean and standard deviation computation times for all the planners under comparison. The pure geometrical methods (AVOCADO and ORCA) share the same inexpensive computational

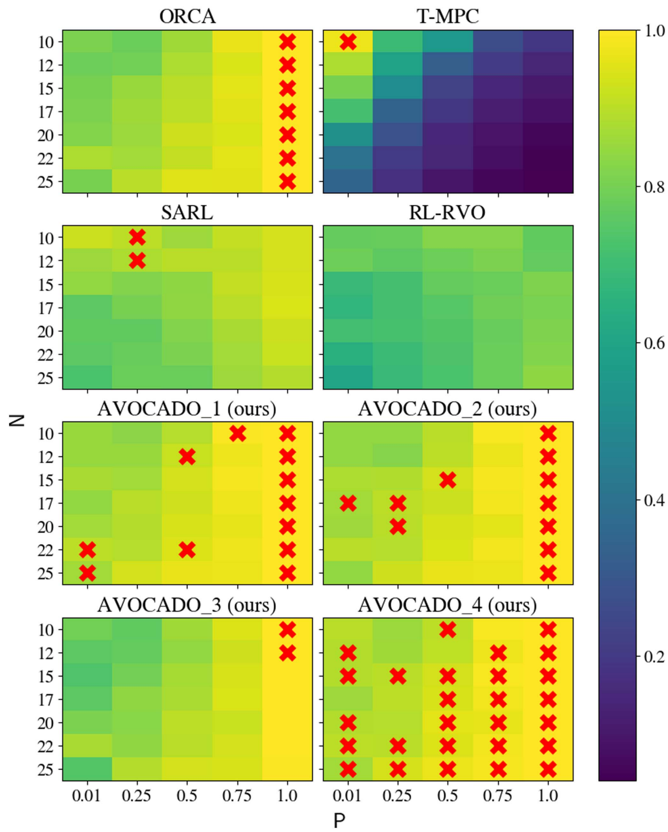


Fig. 12. Success rates of robots with the different planners in *crossing* scenarios. The color bar refers to success rate as defined in Section V-B, whereas P refers to the proportion of cooperative robots. The yellower the better.

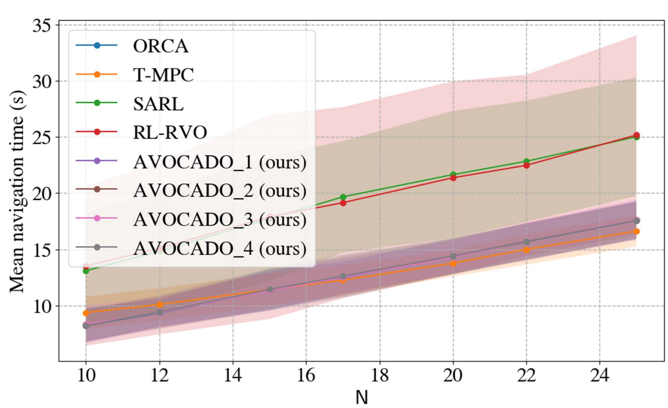


Fig. 13. Mean and standard deviation of navigation times (in seconds) of successful robots for the different planners, gathered in 128 *square* scenarios with $P = 0.5$.

cost, requiring hundreds of milliseconds to compute a solution in crowded environments. The time increases with the number of robots and agents since the number of entities under consideration grow, but this growth is linear with the number of agents and robots within the sensor range. The other methods take, by orders of magnitude, much more time to compute their navigation commands. This computational burden may prevent their use in real hardware applications with constrained

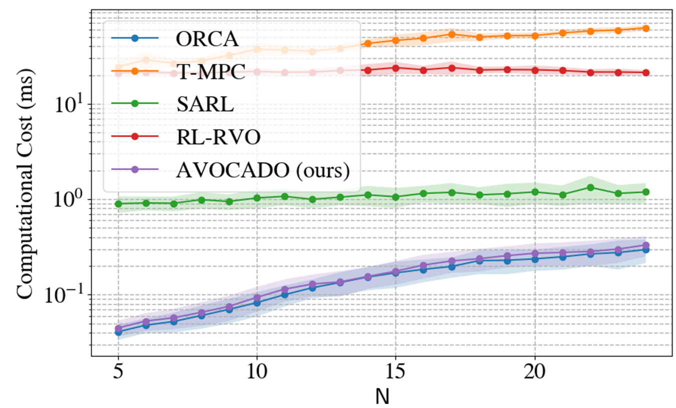


Fig. 14. Mean and standard deviation of the computational times (in milliseconds) for the different planners. *Y*-axis is in a logarithmic scale.

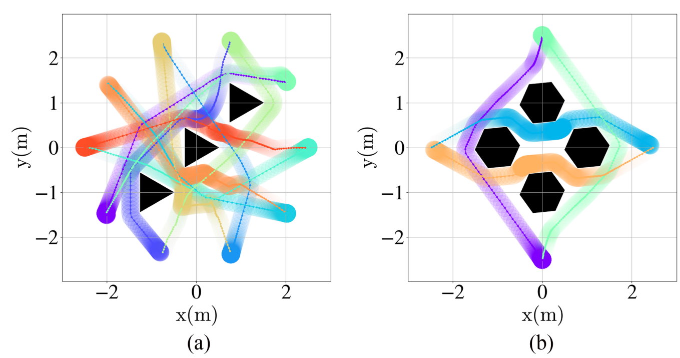


Fig. 15. Multirobot experiments with static obstacles.

resources, especially if there are other higher level tasks that use those resources.

C. Static Obstacles

As real-world scenarios often include other kind of obstacles, we study the behavior of AVOCADO in the presence of static obstacles. We followed the same approach of previous VO-based methods [12], [13], [28], [29], [35], which consider representing the boundaries of the obstacles as segments. Half-spaces constraining the velocities that lead to collision with these segments are directly included in the linear problem optimization. In this way, collision avoidance with static obstacles is naturally achieved with no extra cost.

To verify this insight, we conduct simulated experiments where robots following AVOCADO navigate in a circular scenario that includes static obstacles. Fig. 15 shows that the robots are able to adjust their trajectories to obstacles of different shapes.

Nevertheless, as previously stated, an important remark is that AVOCADO is a planner that locally selects a noncolliding velocity. Therefore, an additional high-level planner would be needed to navigate in the presence of nonconvex obstacles, such as in a maze-like environment.

VI. Experimental Results

After evaluating AVOCADO in simulated environments, in this section we conduct experiments with real ground robots. We use three Turtlebot 2 robotic platforms that use AVOCADO, and up to three pedestrians as external noncooperative agents. The experiments involve 19 pedestrians, 15 of them external to the project associated to this work. The pedestrians, as the robots, have a fixed starting point and final goal for each experiment, but they have no instructions on how to behave and interact with the robots. In this way, the pedestrians decide by themselves their degree of cooperation with the robots and other pedestrians, their own velocity and the trajectory to follow to reach the goal, making each experiment different and unpredictable from the perspective of the robots. The robots use AVOCADO with the same parameterization as the default ones detailed in Table I. The purpose is to prove that AVOCADO presents zero-shot-transfer capabilities.

We conduct the experiments in an arena of 6×6 m. We used an Optitrack Prime^X 13 W system of markers and 12 cameras to localize the robots and the pedestrians. We also design and implement an extended Kalman filter with a constant velocity assumption to track the positions and velocities of all the robots and pedestrians.

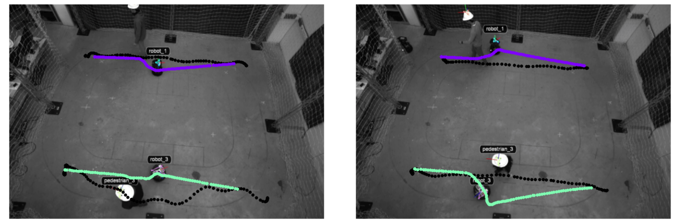
We design three representative experimental scenarios, running them using different combinations of pedestrians, leading to a total of 33 experiments. Videos of the experiments can be found in the supplementary material. The three scenarios are as follows.

1) *Head-on:* two robots are placed next to each other in two corners of the arena, and two pedestrians are placed in the other two corners of the arena, each of them facing one of the robots. The goal of the robots and the pedestrians is to exchange their position with the pedestrian or robot that is in front of them.
2) *Circle:* as in the simulations, pedestrians, and robots are arranged in an evenly spaced circular formation, alternating robots and pedestrians. The goal of all the players is to go to the opposite side of the circle.
3) *Crossing:* two robots are located in the medium point of two opposite sides of the arena, facing each other. In one of the other sides we place a pedestrian and a robot, while two pedestrians are placed in the remaining side of the arena. The team robot–pedestrian and the team pedestrian–pedestrian are told to cross the arena toward the position of the other team. The goal of the first two robots is to cross the perpendicular traffic flow to exchange their position.

Fig. 16 shows two examples of head-on experiments. In the first experiment (a) one of the pedestrians (bottom) decides to evade the robot. AVOCADO detects that and adapts to the situation, following a straight trajectory. Reciprocal dance problem is not observed. The other pedestrian in Fig. 16(a) slightly modifies the trajectory to cooperate with the robot. Fig. 16(b) shows the same experiment with other pedestrians, that decide to not cooperate at all, demonstrating how the robots are able to take all the responsibility to do the collision avoidance maneuver.

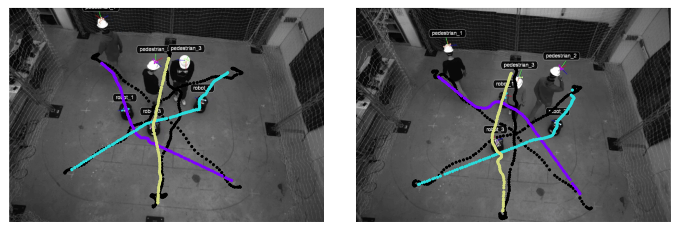
In Fig. 17(a), pedestrians participating in the circle experiment walked with different velocities. When the person starting in the down-right leaves the intersection, the person starting in the top-center is still starting the trajectory. The robots adapt to the different velocities and safely reach the goal with fluid trajectories. The purple robot traverses the intersection moving to its right side, as there is free space there. Fig. 17(b) depicts a situation where the human starting in the down-right and the one starting in the top-center have similar behaviors as in the previous experiment in terms of speed. The pedestrian starting in the bottom-left, however, is faster than its previous homologous. The robots accommodate to this situation by being more cooperative and more reactive (yellow and blue) or choosing a different trajectory in the space that pedestrian is leaving behind (purple).

A standard crossing scenario is represented in Fig. 18(a). All the agents forming the traffic in the middle of the arena respect their lane. The two robots in opposite sides choose a side to cross the intersection, passing between the two nearby agents and the two far ones. The girl starting behind the purple robot overtakes it safely, through the open space that is in their left. The



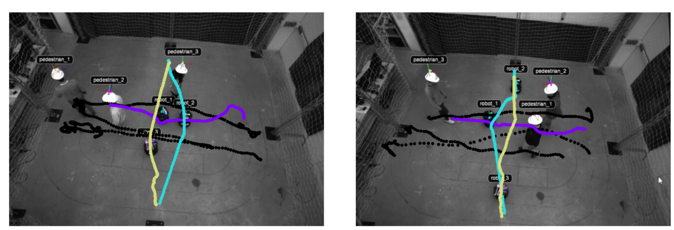
(a) (b)

Fig. 16. Head-on experiment, groups 2 and 4. Two robots face two pedestrians, exchanging their initial position. The pedestrians trajectories are represented in black and the robots with different colors. (a) Group 2. (b) Group 4.



(a) (b)

Fig. 17. Circle experiment, groups 3 and 5. Three robots and three pedestrians, evenly spaced in a circle, exchange their positions with the opposite player. (a) Group 3. (b) Group 5.



(a) (b)

Fig. 18. Crossing experiment, groups 4 and 5. Two robots, in opposite sides, exchange positions by crossing an intersection with one robot and three pedestrians. (a) Group 4. (b) Group 5.

robot collaborates but giving her more space to do it. Fig. 18(b), however, shows a different situation. The pedestrian behind the other pedestrian decides to accelerate and overtake through the central lane. The purple robot has to slow down to evade a frontal collision. This delay in the central traffic makes the blue robot avoid it through the right side, and the yellow adapt to it as it arrives there.

Overall, extensive hardware experiments with multiple robots and humans prove that AVOCADO is effective navigation strategy for collision avoidance in mixed cooperative/noncooperative environments. AVOCADO is a zero-shot approach since no further tuning is required to transfer the algorithm from simulations to real robots. In this sense, AVOCADO preserves the properties observed in simulations in terms of success rate, fluidity, adaptation to unknown degrees of cooperation and computational efficiency.

VII. Conclusion

In this work we have presented AVOCADO, a novel approach to solve the robot collision avoidance problem in mixed cooperative/noncooperative multiagent environments, where the degree of cooperation of the agents is unknown. AVOCADO parameterizes the degree of cooperation of the agents using a scalar value that is adapted in real-time using a novel adaptive law based on nonlinear opinion dynamics. We have shown that the adaptive law, under an appropriate tuning, guarantees robot decision on the degree of cooperation of the agent before collision. To do so, we first exploited the geometry of the problem to develop a novel attention mechanism that depends on the expected time to collision. The attention mechanism is also used to propose a novel method to overcome deadlocks generated by symmetries. The second key to achieve adaptation involved a novel geometrical estimator that predicts the opinion that the agent has on the adapted degree of cooperation of the robot. The estimator only relies on the measured position and velocity of the agent. Finally, AVOCADO integrates the adapted degree of cooperation in a linear program that minimizes the difference between desired and actual velocity while avoiding collision.

Simulated experiments have validated AVOCADO, showing that it is inexpensive to compute and overcomes existing approaches in terms of success rate, number of avoided collisions, computational time and time to reach the goal. In this sense, our solution is readily implementable in low-cost hardware devices, leaving the computational and memory resources free for any desired high-level task such as semantic mapping, autonomous tracking or package delivery. Moreover, extensive experiments with multiple robots and humans have demonstrated that AVOCADO presents zero-shot-transfer capabilities and works in scenarios with uncertainties, noise and evolving unpredictable human behaviors.

As future work, one direction is to explore the integration of AVOCADO in a higher level planner that avoids dense crowded areas or convex obstacles. AVOCADO, since it does not have an horizon planner, sometimes enter in conflicting situations that lead to local minima or unavoidable collisions. Another research direction consists in extending AVOCADO to general robot dynamic models, as other VO-based planners [13], [29]; and to 3-D settings. Finally, it would be interesting to develop a method that, by just relying on onboard sensors, is able to consider how neighboring interactions are affecting the motion of the nearby agents, in order to distinguish noncooperative behaviors stemming from the degree of cooperation of the agents from the ones enforced by their own collision avoidance goals.

References

[1] H. Choset, K. M. Lynch, S. Hutchinson, G. A. Kantor, and W. Burgard, *Principles of Robot Motion: Theory, Algorithms, and Implementations*. Cambridge, MA, USA: MIT Press, 2005.

[2] C. Mavrogiannis et al., “Core challenges of social robot navigation: A survey,” *ACM Trans. Hum.-Robot Interact.*, vol. 12, no. 3, pp. 1–39, 2023.

[3] J. F. Fisac et al., “Probabilistically safe robot planning with confidence-based human predictions,” in *Robotics: Science and Systems*. Cambridge, MA, USA: MIT Press, 2018.

[4] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras, “Human motion trajectory prediction: A survey,” *Int. J. Robot. Res.*, vol. 39, no. 8, pp. 895–935, 2020.

[5] D. Fridovich-Keil et al., “Confidence-aware motion prediction for real-time collision avoidance1,” *Int. J. Robot. Res.*, vol. 39, no. 2-3, pp. 250–265, 2020.

[6] R. Mirsky, X. Xiao, J. Hart, and P. Stone, “Conflict avoidance in social navigation-a survey,” *ACM Trans. Hum.-Robot Interact.*, vol. 13, no. 1, pp. 1–36, 2024.

[7] Y. Tian et al., “Search and rescue under the forest canopy using multiple UAVs,” *Int. J. Robot. Res.*, vol. 39, no. 10-11, pp. 1201–1221, 2020.

[8] W. Tabib, K. Goel, J. Yao, C. Boirum, and N. Michael, “Autonomous cave surveying with an aerial robot,” *IEEE Trans. Robot.*, vol. 38, no. 2, pp. 1016–1032, Apr. 2022.

[9] E. Soria, F. Schiano, and D. Floreano, “Predictive control of aerial swarms in cluttered environments,” *Nature Mach. Intell.*, vol. 3, no. 6, pp. 545–554, 2021.

[10] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, “A survey of motion planning and control techniques for self-driving urban vehicles,” *IEEE Trans. Intell. Veh.*, vol. 1, no. 1, pp. 33–55, Mar. 2016.

[11] Y. Song, A. Romero, M. Müller, V. Koltun, and D. Scaramuzza, “Reaching the limit in autonomous racing: Optimal control versus reinforcement learning,” *Sci. Robot.*, vol. 8, no. 82, 2023, Art. no. eadg1462.

[12] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, “Reciprocal n-body collision avoidance,” in *Proc. Robot. Res.: 14th Int. Symp.*, 2011, pp. 3–19.

[13] J. Alonso-Mora, P. Beardsley, and R. Siegwart, “Cooperative collision avoidance for nonholonomic robots,” *IEEE Trans. Robot.*, vol. 34, no. 2, pp. 404–420, Apr. 2018.

[14] R. Han, S. Chen, and Q. Hao, “Cooperative multi-robot navigation in dynamic environment with deep reinforcement learning,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 448–454.

[15] M. Boldrer, A. Serra-Gomez, L. Lyons, J. Alonso-Mora, and L. Ferranti, “Rule-based Lloyd algorithm for multi-robot motion planning and control with safety and convergence guarantees,” 2023, *arXiv:2310.19511*.

[16] A. Bajcsy et al., “A scalable framework for real-time multi-robot, multi-human collision avoidance,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2019, pp. 936–943.

[17] Q. Li, F. Gama, A. Ribeiro, and A. Prorok, “Graph neural networks for decentralized multi-robot path planning,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 11785–11792.

[18] A. Patwardhan, R. Murai, and A. J. Davison, “Distributing collaborative multi-robot planning with Gaussian belief propagation,” *IEEE Robot. Automat. Lett.*, vol. 8, no. 2, pp. 552–559, Feb. 2023.

[19] S. Zhang, K. Garg, and C. Fan, “Neural graph control barrier functions guided distributed collision-avoidance multi-agent control,” in *Proc. Conf. Robot Learn.*, 2023, pp. 2373–2392.

[20] P. Long, W. Liu, and J. Pan, “Deep-learned collision avoidance policy for distributed multiagent navigation,” *IEEE Robot. Automat. Lett.*, vol. 2, no. 2, pp. 656–663, Apr. 2017.

[21] M. Everett, Y. F. Chen, and J. P. How, “Collision avoidance in pedestrian-rich environments with deep reinforcement learning,” *IEEE Access*, vol. 9, pp. 10357–10377, 2021.

[22] B. Brito, B. Floor, L. Ferranti, and J. Alonso-Mora, “Model predictive contouring control for collision avoidance in unstructured dynamic environments,” *IEEE Robot. Automat. Lett.*, vol. 4, no. 4, pp. 4459–4466, Oct. 2019.

[23] S. Poddar, C. Mavrogiannis, and S.S. Srinivasa, "From crowd motion prediction to robot navigation in crowds," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2023, pp. 6765–6772.

[24] D. Morilla-Cabello, L. Mur-Labadia, R. Martinez-Cantin, and E. Montijano, "Robust fusion for Bayesian semantic mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2023, pp. 76–81.

[25] A. Asgharivaskasi and N. Atanasov, "Semantic OcTree mapping and Shannon mutual information computation for robot exploration," *IEEE Trans. Robot.*, vol. 39, no. 3, pp. 1910–1928, Jun. 2023.

[26] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using the relative velocity paradigm," in *Proc. IEEE Int. Conf. Robot. Automat.*, 1993, pp. 560–565.

[27] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *Int. J. Robot. Res.*, vol. 17, no. 7, pp. 760–772, 1998.

[28] J. Van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2008, pp. 1928–1935.

[29] J. Alonso-Mora, A. Breitenmoser, M. Rufli, P. Beardsley, and R. Siegwart, "Optimal reciprocal collision avoidance for multiple non-holonomic robots," in *Proc. Distrib. Auton. Robot. Syst.: 10th Int. Symp.*, Springer, 2013, pp. 203–216.

[30] O. Brock and O. Khatib, "High-speed navigation using the global dynamic window approach," in *Proc. IEEE Int. Conf. Robot. Automat.*, 1999, pp. 341–346, vol. 1.

[31] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Int. J. Robot. Res.*, vol. 5, no. 1, pp. 90–98, 1986.

[32] S. Petti and T. Fraichard, "Safe motion planning in dynamic environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2005, pp. 2210–2215.

[33] G. Ferrer, A. Garrell, and A. Sanfeliu, "Robot companion: A social-force based approach with human awareness-navigation in crowded environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 1688–1694.

[34] F. Vesentini, R. Muradore, and P. Fiorini, "A survey on velocity obstacle paradigm," *Robot. Auton. Syst.*, vol. 174, 2024, Art. no. 104645.

[35] M. Rufli, J. Alonso-Mora, and R. Siegwart, "Reciprocal collision avoidance with motion continuity constraints," *IEEE Trans. Robot.*, vol. 29, no. 4, pp. 899–912, Aug. 2013.

[36] D. Bareiss and J. Van den Berg, "Generalized reciprocal collision avoidance," *Int. J. Robot. Res.*, vol. 34, no. 12, pp. 1501–1514, 2015.

[37] K. Guo, D. Wang, T. Fan, and J. Pan, "VR-ORCA: Variable responsibility optimal reciprocal collision avoidance," *IEEE Robot. Automat. Lett.*, vol. 6, no. 3, pp. 4520–4527, Jul. 2021.

[38] J. Qin, J. Qin, J. Qiu, Q. Liu, M. Li, and Q. Ma, "SRL-ORCA: A socially aware multi-agent mapless navigation algorithm in complex dynamic scenes," *IEEE Robot. Automat. Lett.*, vol. 9, no. 1, pp. 143–150, Jan. 2024.

[39] L. Tai, J. Zhang, M. Liu, and W. Burgard, "Socially compliant navigation through raw depth inputs with generative adversarial imitation learning," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 1111–1117.

[40] A. Pokle et al., "Deep local trajectory replanning and control for robot navigation," in *Proc. 2019 Int. Conf. Robot. Automat.*, 2019, pp. 5815–5822.

[41] Z. Xie, P. Xin, and P. Dames, "Towards safe navigation through crowded dynamic environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 4934–4940.

[42] E. Sebastián, T. Duong, N. Atanasov, E. Montijano, and C. Sagüés, "LEMURS: Learning distributed multi-robot interactions," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 7713–7719.

[43] P. Long, T. Fan, X. Liao, W. Liu, H. Zhang, and J. Pan, "Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 6252–6259.

[44] T. Fan, P. Long, W. Liu, and J. Pan, "Distributed multi-robot collision avoidance via deep reinforcement learning for navigation in complex scenarios," *Int. J. Robot. Res.*, vol. 39, no. 7, pp. 856–892, 2020.

[45] R. Ourari, K. Cui, A. Elshamanhory, and H. Koeppl, "Nearest-neighbor-based collision avoidance for quadrotors via reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2022, pp. 293–300.

[46] K. Cui, M. Li, C. Fabian, and H. Koeppl, "Scalable task-driven robotic swarm control via collision avoidance and learning mean-field control," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 1192–1199.

[47] D. Martinez-Baselga, L. Riazuelo, and L. Montano, "Improving robot navigation in crowded environments using intrinsic rewards," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 9428–9434.

[48] E. Sebastian, T. Duong, N. Atanasov, E. Montijano, and C. Sagues, "Physics-informed multi-agent reinforcement learning for distributed multi-robot problems," 2023, *arXiv:2401.00212*.

[49] B. Brito, M. Everett, J. P. How, and J. Alonso-Mora, "Where to go next: Learning a subgoal recommendation policy for navigation in dynamic environments," *IEEE Robot. Automat. Lett.*, vol. 6, no. 3, pp. 4616–4623, Jul. 2021.

[50] H. Yu, C. Hirayama, C. Yu, S. Herbert, and S. Gao, "Sequential neural barriers for scalable dynamic obstacle avoidance," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2023, pp. 11241–11248.

[51] R. Han et al., "Reinforcement learned distributed multi-robot navigation with reciprocal velocity obstacle shaped rewards," *IEEE Robot. Automat. Lett.*, vol. 7, no. 3, pp. 5896–5903, Jul. 2022.

[52] L. Wang, A. D. Ames, and M. Egerstedt, "Safety barrier certificates for collisions-free multirobot systems," *IEEE Trans. Robot.*, vol. 33, no. 3, pp. 661–674, Jun. 2017.

[53] D. Morgan, S.-J. Chung, and F. Y. Hadaegh, "Model predictive control of swarms of spacecraft using sequential convex programming," *J. Guid., Control, Dyn.*, vol. 37, no. 6, pp. 1725–1740, 2014.

[54] M. Althoff, G. Frehse, and A. Girard, "Set propagation techniques for reachability analysis," *Annu. Rev. Control Robot., Auton. Syst.*, vol. 4, pp. 369–395, 2021.

[55] M. Chen, J. C. Shih, and C. J. Tomlin, "Multi-vehicle collision avoidance via Hamilton-Jacobi reachability and mixed integer programming," in *Proc. IEEE Conf. Decis. Control*, 2016, pp. 1695–1700.

[56] S. Bansal and C. J. Tomlin, "Deepreach: A deep learning approach to high-dimensional reachability," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 1817–1824.

[57] K. D. Julian and M. J. Kochenderfer, "Reachability analysis for neural network aircraft collision avoidance systems," *J. Guid., Control, Dyn.*, vol. 44, no. 6, pp. 1132–1142, 2021.

[58] A. Li, L. Sun, W. Zhan, M. Tomizuka, and M. Chen, "Prediction-based reachability for collision avoidance in autonomous driving," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 7908–7914.

[59] K. Long, V. Dhiman, M. Leok, J. Cortés, and N. Atanasov, "Safe control synthesis with uncertain dynamics and constraints," *IEEE Robot. Automat. Lett.*, vol. 7, no. 3, pp. 7295–7302, Jul. 2022.

[60] D. Panagou, D. M. Stipanović, and P. G. Voulgaris, "Distributed coordination control for multi-robot networks using Lyapunov-like barrier functions," *IEEE Trans. Autom. Control*, vol. 61, no. 3, pp. 617–632, Mar. 2016.

[61] Y. Chen, A. Singletary, and A. D. Ames, "Guaranteed obstacle avoidance for multi-robot operations with limited actuation: A control barrier function approach," *IEEE Control Syst. Lett.*, vol. 5, no. 1, pp. 127–132, Jan. 2021.

[62] J. Schulman et al., "Motion planning with sequential convex optimization and convex collision checking," *Int. J. Robot. Res.*, vol. 33, no. 9, pp. 1251–1270, 2014.

[63] K. Ryu and N. Mehr, "Integrating predictive motion uncertainties with distributionally robust risk-aware control for safe robot navigation in crowds," in *Proc. 2024 IEEE Int. Conf. Robot. Automat.*, 2024, pp. 2410–2417.

[64] O. de Groot, L. Ferranti, D. M. Gavrila, and J. Alonso-Mora, "Scenario-based motion planning with bounded probability of collision," *Int. J. Robot. Res.*, 2025, doi: 10.1177/02783649251315203.

[65] O. de Groot, L. Ferranti, D. M. Gavrila, and J. Alonso-Mora, "Topology-driven parallel trajectory optimization in dynamic environments," *IEEE Trans. Robot.*, vol. 41, pp. 110–126, 2025.

[66] K. D. Katyal, G. D. Hager, and C.-M. Huang, "Intent-aware pedestrian prediction for adaptive crowd navigation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 3277–3283.

[67] S. Liu et al., "Intention aware robot crowd navigation with attention-based interaction graph," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 12015–12021.

[68] C. Mavrogiannis, K. Balasubramanian, S. Poddar, A. Gandra, and S.S. Srinivasa, "Winding through: Crowd navigation via topological invariance," *IEEE Robot. Automat. Lett.*, vol. 8, no. 1, pp. 121–128, Jan. 2023.

[69] C. Altafini, "Consensus problems on networks with antagonistic interactions," *IEEE Trans. Autom. Control*, vol. 58, no. 4, pp. 935–946, Apr. 2013.

[70] P. Jia, A. MirTabatabaei, N. E. Friedkin, and F. Bullo, "Opinion dynamics and the evolution of social power in influence networks," *SIAM Rev.*, vol. 57, no. 3, pp. 367–397, 2015.

[71] P. Cisneros-Velarde, K. S. Chan, and F. Bullo, "Polarization and fluctuations in signed social networks," *IEEE Trans. Autom. Control*, vol. 66, no. 8, pp. 3789–3793, Aug. 2021.

[72] A. Bizyaeva, A. Franci, and N. E. Leonard, "Nonlinear opinion dynamics with tunable sensitivity," *IEEE Trans. Autom. Control*, vol. 68, no. 3, pp. 1415–1430, Mar. 2023.

[73] N. E. Leonard, A. Bizyaeva, and A. Franci, “Fast and flexible multiagent decision-making,” *Annu. Rev. Control Robot. Auton. Syst.*, vol. 7, pp. 12–1, 2024.
[74] N. E. Leonard, K. Lipsitz, A. Bizyaeva, A. Franci, and Y. Lelkes, “The nonlinear feedback dynamics of asymmetric political polarization,” *Proc. Nat. Acad. Sci.*, vol. 118, no. 50, 2021, Art. no. e2102149118.
[75] M. Ordorica Arango, A. Bizyaeva, S. A. Levin, and N. Ehrich Leonard, “Opinion-driven risk perception and reaction in SIS epidemics,” 2024, *arXiv:2410.12993v2*.
[76] S. Park, A. Bizyaeva, M. Kawakatsu, A. Franci, and N. E. Leonard, “Tuning cooperative behavior in games with nonlinear opinion dynamics,” *IEEE Control Syst. Lett.*, vol. 6, pp. 2030–2035, 2022.
[77] H. Hu, K. Nakamura, K.-C. Hsu, N. E. Leonard, and J. F. Fisac, “Emergent coordination through game-induced nonlinear opinion dynamics,” in *Proc. IEEE Conf. Decis. Control*, 2023, pp. 8122–8129.
[78] H. Hu, J. DeCastro, D. Gopinath, G. Rosman, N. E. Leonard, and J. F. Fisac, “Think deep and fast: Learning neural nonlinear opinion dynamics from inverse dynamic games for split-second interactions,” 2024, *arXiv:2406.09810*.
[79] A. Bizyaeva, G. Amorim, M. Santos, A. Franci, and N. E. Leonard, “Switching transformations for decentralized control of opinion patterns in signed networks: Application to dynamic task allocation,” *IEEE Control Syst. Lett.*, vol. 6, pp. 3463–3468, 2022.
[80] C. Cathcart, M. Santos, S. Park, and N. E. Leonard, “Proactive opinion-driven robot navigation around human movers,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2023, pp. 4052–4058.
[81] G. Amorim, A. Bizyaeva, A. Franci, and N. E. Leonard, “Spatially-invariant opinion dynamics on the circle,” *IEEE Control Syst. Lett.*, vol. 8, pp. 3231–3236, 2024.
[82] T. H. Chung, G. A. Hollinger, and V. Isler, “Search and pursuit-evasion in mobile robotics: A survey,” *Auton. Robots*, vol. 31, pp. 299–316, 2011.
[83] E. Sebastián, E. Montijano, and C. Sagüés, “Adaptive multirobot implicit control of heterogeneous herds,” *IEEE Trans. Robot.*, vol. 38, no. 6, pp. 3622–3635, Dec. 2022.
[84] T. Eppenberger, G. Cesari, M. Dymczyk, R. Siegwart, and R. Dubé, “Leveraging stereo-camera data for real-time dynamic obstacle detection and tracking,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 10528–10535.
[85] H. Dong, C.-Y. Weng, C. Guo, H. Yu, and I.-M. Chen, “Real-time avoidance strategy of dynamic obstacles via half model-free detection and tracking with 2D LiDAR for mobile robots,” *IEEE/ASME Trans. Mechatron.*, vol. 26, no. 4, pp. 2215–2225, Aug. 2021.
[86] T. Battisti and R. Muradore, “A velocity obstacles approach for autonomous landing and teleoperated robots,” *Auton. Robots*, vol. 44, no. 2, pp. 217–232, 2020.
[87] C. Chen, Y. Liu, S. Kreiss, and A. Alahi, “Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning,” in *Proc. IEEE Int. Conf. Robot. Automat.*, 2019, pp. 6015–6022.

Diego Martinez-Baselga (Graduate Student Member, IEEE) received the B.Eng. degree in computer science and the M.Sc. degree in robotics, graphics and computer vision from the University of Zaragoza, Zaragoza, Spain, in 2020 and 2022, respectively, where he is working toward the Ph.D. degree in computer science and systems engineering.

In 2023, he was a Visiting Researcher with Autonomous Multi-Robots Lab, TU Delft, Delft, The Netherlands. His research interests include motion planning and collision avoidance using learning and control methods, and multirobot systems.

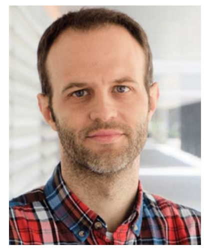
Eduardo Sebastián (Graduate Student Member, IEEE) received the B.Eng. (Hons.) degree in electronic and automatic engineering and the M.Eng. (Hons.) degree in electronics from the University of Zaragoza, Zaragoza, Spain, in 2019 and 2020, respectively, where he is currently working toward the Ph.D. degree in computer science and systems engineering.

In 2022 and 2024, he was a Visiting Scholar with the University of California San Diego, San Diego, CA, USA. His research interests include nonlinear control and distributed multirobot systems.

Dr. Sebastián is a Fulbright Scholar and a DAAD AInet Fellow.

Eduardo Montijano (Member, IEEE) received the M.Sc. degree in computer science and the Ph.D. degree in robotics and systems engineering from the University of Zaragoza, Zaragoza, Spain, in 2008 and 2012, respectively.

From 2012 to 2016, he was a Faculty Member with Centro Universitario de la Defensa, Zaragoza, Spain. He is currently an Associate Professor with the Departamento de Informática e Ingeniería de Sistemas, University of Zaragoza. His research focuses on the design of scene understanding algorithms for planning and control of multiple robots.



Luis Riazuelo (Member, IEEE) received the M.Sc. degree in computer science and M.Eng. degree in biomedical engineering from the University of Zaragoza, Zaragoza, Spain, in 2006 and 2008, respectively, and the Ph.D. degree in computer science from the University of Zaragoza, in 2018.

He is currently an Associate Professor with Computer Science and Systems Engineering Department, University of Zaragoza. Since 2007, he has been a Researcher with Robotics, Perception and Real Time Group, Aragòn Engineering Research. His research interests include scene understanding for mobile robotics in underground environments and topological semantic mapping and localization in intracorporeal medical scenes.

Carlos Sagüés (Senior Member, IEEE) received the M.Sc. degree in computer science and systems engineering and the Ph.D. degree in industrial engineering from the University of Zaragoza, Zaragoza, Spain, in 1989 and 1992, respectively.

In 1994, he was an Associate Professor with the Departamento de Informatica e Ingenieria de Sistemas, University of Zaragoza, where he became a Full Professor in 2009 and also the Head Teacher. He was engaged in research on force and infrared sensors for robots. His research interests include control systems and industry applications, computer vision, visual control, and multivehicle cooperative control.

Luis Montano (Senior Member, IEEE) received the industrial engineering degree and the Ph.D. degree in industrial engineering from the University of Zaragoza, Zaragoza, Spain, in 1981 and 1987, respectively.

He is currently a Full Professor with the University of Systems Engineering and Automation, University of Zaragoza. He is Principal Researcher of the Robotics, Computer Vision and Artificial Intelligence Research Group of the Institute. He was a Director with the Department of Computer Science and Systems Engineering and Deputy Director of the Aragon Engineering Research Institute, University of Zaragoza. His research interests include robotics which include planning and navigation in dynamic environments and multirobot systems.